SUPER 6

n° 6 Dicembre 1984

L. 7.500

Spedizione in abbonamento postale Gruppo III/70

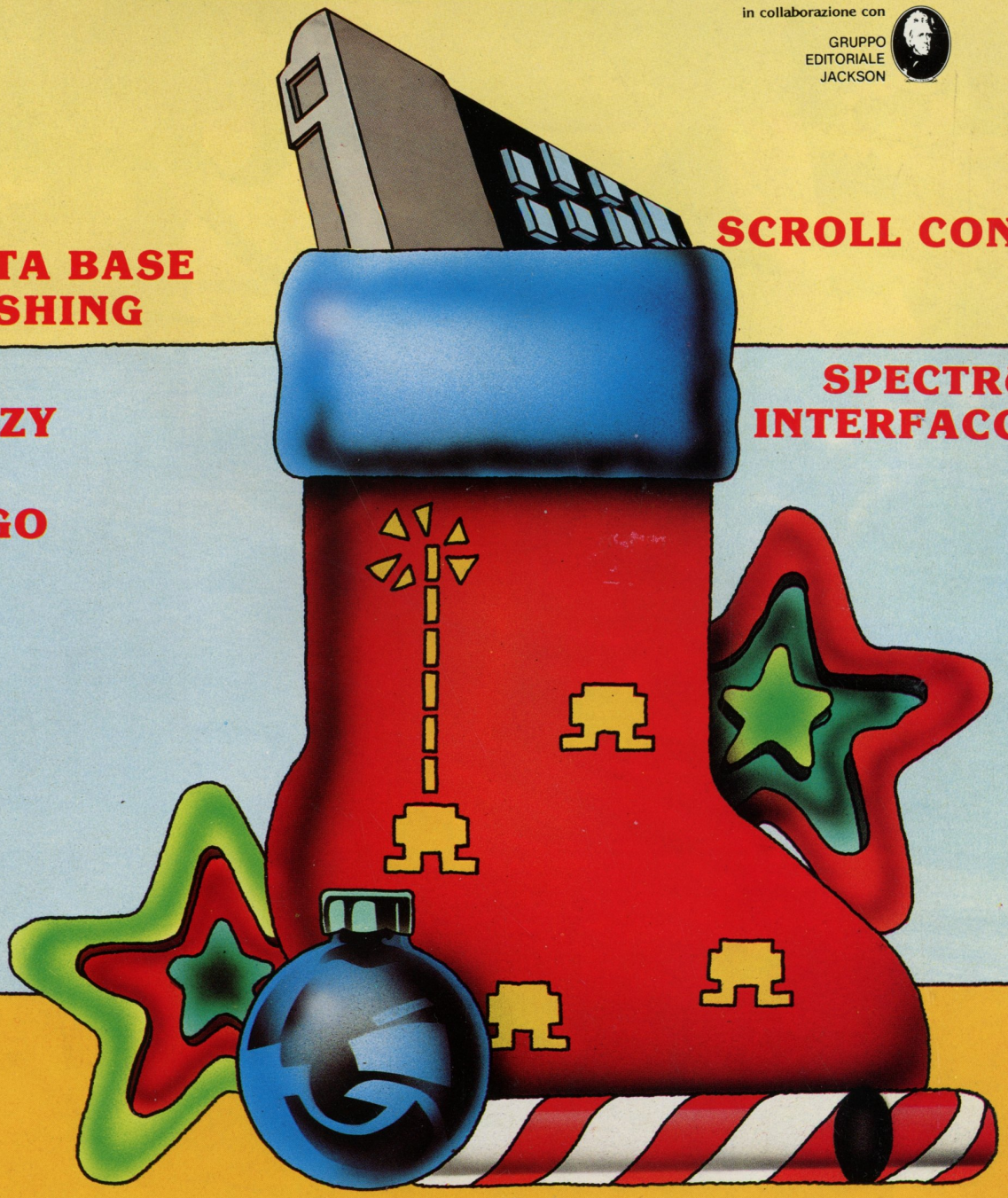
SUPERSINC
la rivista per gli utenti dei prodotti SINCLAIR

Una pubblicazione della

J.soft EDITRICE

in collaborazione con

GRUPPO EDITORIALE JACKSON



SCROLL CONTROL

DATA BASE HASHING

SPECTRUM E INTERFACCIA 1

GIZZY

LOGO

E' IN EDICOLA

Bit

Annuario hardware 1985

tutto l'hardware
per l'informatica
in Italia

Una realizzazione
GRUPPO EDITORIALE JACKSON
e
Istituto SISDOBDA

Home Computer
Micro e Personal Computer
Minisistemi - Supermini - Stampanti - Plotter

Supplemento a Bit Nr. 55/Novembre 1984 - Lire 8.000

SUPER sinc

5 EDITORIALE POINT

RIFLESSIONI
di **Pietro Dell'Orco**

7 READ &WRITE

LA POSTA DEI LETTORI

9 OPEN

SPECTRUM E INTERFACCIA 1: UN SISTEMA COMPLETO
di **Marcello Spero**

13 LA CULTURA IF INFORMATICA ...then

LOGO E TARTARUGHE
di **Tim Hartenell**
trad. e adatt. di
Marcello Spero

21 REM:HW

COME REALIZZARE UN'INTERFACCIA
di **Marcello Spero**

29 LOAD

29 GIZZY
di **Angelo Motta**

35 PALLONZOLO
di **Daniele Malavasi**

44 SCROLL CONTROL
di **Emilio Roncoroni**

47 DATA BASE HASHING
di **Antonio Motta**

53 SINDISC 01
di **Daniele Malavasi**

64 FLATMAN
di **J. Dave Rogers**
trad. e adatt. di
Angelo Motta

68 LA CHIAVE MAGICA
di **Raffaele Ilardo**

74 R·U·N

RECENSIONE LIBRI
a cura di **Lucio Bragagnolo**

77 ZX CLUB

I CLUB SINCLAIR

81 input·output

PICCOLI ANNUNCI

J.soft s.r.l
DIREZIONE, REDAZIONE, AMMINISTRAZIONE
Via Rosellini, 12
20124 MILANO
Tel. (02) 68.88.228-68.37.97
68.80.841/2/3

DIRETTORE RESPONSABILE:
Pietro Dell'Orco

COORDINAMENTO TECNICO:
Riccardo Paolillo

REDAZIONE:
Lucio Bragagnolo
Marcello Spero

HANNO COLLABORATO A QUESTO NUMERO
Raffaele Ilardo
Daniele Malavasi
Angelo Motta
Antonio Motta
Emilio Roncoroni

GRAFICA E IMPAGINAZIONE:
Margherita La Noce
Ivana Rossi
Raffaella Toffolatti

FOTOCOMPOSIZIONE:
d&b Via Vignola, 5
Tel. 02/59.85.08
20133 MILANO

CONTABILITÀ:
Giulia Pedrazzini
Flavia Bonaiti

AUTORIZZAZIONE ALLA PUBBLICAZIONE:
Tribunale di Milano n° 201
del 14.04.1984

STAMPA:
Litografia del Sole
Albairate (MI)

PUBBLICITÀ
Concessionaria per l'Italia e
J. Advertising s.r.l.
Viale Restelli, 5
20124 Milano
Tel. (02) 6882895-6880606-6887233
Tlx. 316213 REINA 1

Concessionaria esclusivo per la
DIFFUSIONE in Italia e Estero:
SODIP - Via Zuretti, 25
20125 MILANO

Spedizione in abbonamento
postale Gruppo III/70
Prezzo della rivista L. 7.500
Numero arretrato L. 15.000
Abbonamento annuo: L. 82.500
(11 numeri con cassetta) L. 38.500
(11 numeri senza cassetta) per l'estero:
L. 110.000 (11 numeri con cassetta)
L. 50.000 (11 numeri senza cassetta)
i versamenti devono essere indirizzati a:
J.soft s.r.l.
Via Rosellini, 12
20124 MILANO
mediante emissione di assegno
bancario, cartolina, vaglia o
utilizzando il c/c postale
n. 19445204
Per i cambi di indirizzo indicare,
oltre al nuovo, anche l'indirizzo
precedente ed allegare alla
comunicazione l'importo di L. 500,
anche in francobolli

© TUTTI I DIRITTI DI
RIPRODUZIONE O TRADUZIONE
DEGLI ARTICOLI PUBBLICATI
SONO RISERVATI

GLI ARTICOLI TRADOTTI SONO
TRATTI DALLA RIVISTA
ZX COMPUTING - ARGUS
SPECIALIST
PUBLICATION L.T.D.
1 GOLDEN SQUARE LONDON
W1R3AB

USPI Rivista associata all'Unione Stampa Periodica Italiana

J.soft®

CERCA:

ambosessi di qualsiasi età, residenti in Italia o all'estero

REQUISITI RICHIESTI:

forte interesse per gli home computer Commodore o Sinclair disponibilità tempo libero per appassionante lettura di "superivista" dedicata

OFFRE:

abbonamento a 11 numeri di SuperSinc
al **prezzo speciale** di
L. 66.000
compresa cassetta
oppure
L. 30.000
per la sola rivista

inviando il coupon pubblicato a fondo pagina otterrete pronta soddisfazione alla vostra richiesta.

Abbonarsi è semplice! Effettuate il versamento con l'apposito modulo c.c.p. inserito in questo fascicolo, oppure ritagliate il tagliando abbonamenti pubblicato in questa pagina e speditelo allegando un assegno intestato a: *J. soft* - Via Rosellini 12 - 20124 Milano.

Tagliando abbonamento a SUPERSINC da inviare in busta chiusa a: J.*soft* - Via Rosellini 12 - 20124 Milano

☐ Abbonamento a 11 numeri di SUPERSINC al prezzo speciale di L. 30.000

☐ Abbonamento a 11 numeri di SUPERSINC + cassetta con tutti i programmi pubblicati al prezzo speciale di L. 66.000

cognome _____ nome _____

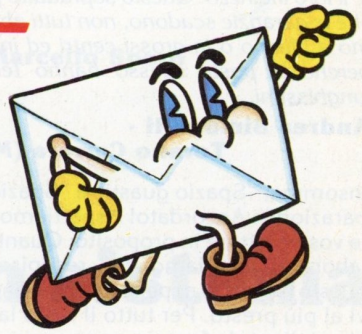
via _____ città _____

cap. _____ provincia _____ data _____

firma _____

READ &WRITE

La posta dei lettori



Cari lettori,
riceviamo continuamente telefonate e lettere di autori di programmi spediti a SUPERSINC, alla ricerca di informazioni circa il destino delle loro creature. Possiamo dirvi solamente, a questo proposito, di... avere pazienza! Siamo veramente sommersi dai vostri contributi, tanto che a volte è difficile perfino esaminarli, per mancanza di tempo. In ogni caso, vi garantiamo che tutti i vostri lavori saranno (sono!) attentamente valutati e che a tutti verrà data una risposta il più esauriente possibile.
Anche voi, a questo proposito, potete darci una mano, principalmente aiutandoci a velocizzare il lavoro. Come? Presto detto: agevolandoci inviando programmi ben documentati, il più possibile a prova di errore, completi di tutti i vostri dati (compresi luogo e data di nascita) e unendo a qualunque routine in linguaggio macchina (o il set di caratteri grafici, o gli array di qualsiasi tipo) il corrispondente programma generatore scritto in BASIC, per comodità di listaggio.
E ora, a voi!

Funziona davvero!

Ho trovato una serie di POKE che consentono di eliminare la maggior parte dei mostri presenti nelle stanze di "Jet Set Willy". La procedura da seguire è questa: NON caricate il primo blocco del programma, sostituendo ad esso le seguenti righe:

```
10 CLEAR 25000: LOAD ""CODE
20 FOR a=43780 TO 45823: POKE
   a,0: NEXT a
30 FOR a=46080 TO 49151: POKE
   a,0: NEXT a
40 POKE 35899,0
50 BORDER 1
60 RANDOMIZE USR 33792
```

date RUN e caricate "Jet Set", ripeto, SENZA caricarne il primo blocco in BASIC! E... buona esplorazione a tutti!

Roberta Quirico
Novate Milanese (MI)

Nulla da dire: abbiamo verificato che la cosa funziona davvero. Anche così "Jet Set Willy" non diventa certo facile da risolvere, ma la quasi totale assenza di mostri e ostacoli in movimento lo rende un pochino meno ostico, e forse anche più piacevole.

E la cassetta?

Gentilissima redazione di SUPERSINC, sono un ragazzo di 15 anni che, avendo acquistato da poco lo SPECTRUM, non possiede ancora la stampante.
Avendo realizzato un programma che permette di giocare a Master Mind contro il computer, ho pensato di mandarvi ugualmente il listato, battuto a macchina.
Approfitto della presente anche per suggerirvi di pubblicare qualche lezione riguardante il BASIC Sinclair e anche il complicatissimo linguaggio macchina che non ho ancora imparato ad usare.

Walter Luppino - (Torino)

Caro Walter,
innanzitutto grazie per complimenti e auguri (che non riportiamo perché siamo molto timidi...)! Quanto al tuo programma, non è male: l'unico problema è però che non ce lo hai mandato su cassetta, così che non potremo utilizzarlo, per mancanza del tempo necessario a ribatterlo. La prossima volta ricordati di spedirci anche la cassetta, OK? Quanto al linguaggio macchina, ne riparleremo.
Intanto, per imparare meglio il BASIC, potrebbe essere una buona idea quella di studiare accuratamente, magari provando a pasticciarli un po', i programmi da noi pubblicati, non credi? Siamo sicuri che potresti ricavarne un sacco di conoscenze utili!

Hardware, software, Microdrive, ROM...

Sarebbero interessanti articoli sull'hardware, sul contenuto software della ROM e sul software per la gestione dei file su Microdrive...

Marco Pernigoni - (Milano)

Ci stiamo già pensando, e non è detto che non venga fuori qualcosa di concreto in tempi brevi...

A caccia di bug

Spettabile direzione,
perché non includete nella vostra rivista uno spazio dedicato a curiosità, particolarità, bug dei giochi in commercio?

Marco Carvana - (Genova)

Quando troviamo qualcosa che valga la pena di essere evidenziato lo pubblichiamo sempre "al volo"; contiamo però moltissimo, a questo proposito, sulla collaborazione di tutti. Intanto, godetevi la lettera riguardante "Jet Set Willy", più sopra...

Questi guasti, che "rotture"...

Sono possessore di uno ZX Spectrum 48K con interfaccia 1, due microdrive, stampante Seikosha 500A, altre interfacce ed il monitor b/n. Purtroppo alcuni giorni fa il computer è andato in palla, ignora la tastiera, emette il suono di memoria piena e, anche staccando l'alimentazione per qualche secondo, al reinserimento dello spinotto il monitor rimane buio. Lo Spectrum è senz'altro una buona macchina, ma mi sembra molto strano che a nessuno dei suoi utenti sia mai capitato un guasto.
Questo è quanto io proporrei di inserire nella rivista.
Analisi commentata della macchina e relative interfacce, anche di marche diverse, ma che funzionino su di essa.
Uno spazio riservato a chi ha avuto dei guasti, per commentarli ed eventualmente indicare i componenti sostituiti (cosa che potrebbe esserre chiesta al tecnico che ha eseguito la riparazione). Infine, la pubblicazione degli schemi elettrici con riportate tensione e forme d'onda.
Penso che tutto questo interesserebbe parecchia gente, in special modo quelli che hanno anche la passione per l'elettronica.
Ultima cosa, e spero che non sia la fati-

dica goccia, chiedere a tutti i laboratori in grado di riparare lo Spectrum di inviare il loro indirizzo. Questo soprattutto perché le garanzie scadono, non tutti abitiamo a Milano o in grossi centri ed infine perché le poste spesso hanno tempi lunghissimi.

Andrea Simonelli - Torano Carrara (MS)

Insomma, "Spazio guasti" e "Spazio riparazioni". Accordato! Aspettiamo ora le vostre lettere in proposito. Quanto ai laboratori speriamo che recepiscano questo (sensato) appello e si facciano vivi al più presto. Per tutto il resto la cosa migliore da fare rimane seguire SUPERSINC. e vedrà che un poco alla volta avrà tutto ciò che desidera.

Diavolo, che stampante!

Gentile SUPERSINC,
desidererei sapere con quale stampante eseguite i vostri listati.

Giovanni Targa - (Bolzano)

I listati ZX81, per ora, sono ottenuti da una comune Seikosha GP50S, mentre lo Spectrum è interfacciato ad una - meno comune - Diablo 630, specificatamente disegnata per lavorare col Personal Computer IBM, ma che funziona egregiamente, grazie a un apposito programma, anche con il "nostro".

Resetto, ergo non carico

Sono un felice possessore di uno ZX Spectrum 16K. Vi scrivo per avere una delucidazione: ho avuto modo di notare, varie volte, che alcuni programmi per lo Spectrum 16K al termine del caricamento si "autoresettano" per cause a me ignote (se no, perché vi scriverei?).
Sapete per caso di cosa possa trattarsi? Una cosa è certa: il fenomeno non dipende dal mio computer né dal registratore, né dalla cassetta, dato che la stessa cosa accade ad altri miei amici, possessori di Spectrum 16K, con cassette diverse. Sullo Spectrum 48K, invece, tutto OK!!! Non rispondetemi dicendo che il programma occupa più di 16K di memoria, perché so con certezza che non è così. Non potrebbe dipendere dal fatto che il gioco sia stato elaborato o modificato su uno Spectrum 48K?
Si può rimediare con un'istruzione diretta che modifichi il valore della RAMTOP o altro?
Vi prego, rispondetemi: sono disperato! P.S. quasi dimenticavo: quale dovrebbe essere il messaggio che alla fine o durante il caricamento indica la mancanza di memoria sufficiente?

Paolo

Caro Paolo,
avresti anche potuto mettere il tuo indirizzo, ma per questa volta ti risponderemo ugualmente, purtroppo con notizie non proprio consolanti.
Come abbiamo già brevemente accennato in uno dei numeri scorsi, gli Spectrum finora usciti dagli stabilimenti Sinclair non sono tutti uguali, tanto che ogni versione è stata contrassegnata da un numero diverso. I primi Spectrum giunti in Italia appartenevano alla versione 2; successivamente sono stati introdotti anche i "versione 3", che differiscono dai precedenti nei valori ritornati da IN, la funzione di lettura tastiera che permette allo ZX di "vedere" la pressione simultanea di più tasti, diversamente da ciò che accade con INKEY$. Infine sono arrivate le versioni 3a e 3b, che danno luogo all'inconveniente di cui ci parli. Purtroppo, non ci risulta ci siano altri sistemi di rimediare al problema diversi dall'...espansione a 48K, che funziona invece perfettamente.
Quasi dimenticavamo: il messaggio di errore a cui ti riferisci è "Out of memory", e compare una volta terminato il caricamento dell'header (contenente fra le altre cose l'indicazione della lunghezza del programma) o durante il funzionamento del programma stesso. Il tutto, sia ben chiaro, parlando di programmi BASIC convenzionali: il software scritto in linguagguo macchina (e la stragrande maggioranza dei programmi commerciali è realizzata in questo modo), quando non preveda espressamente questa situazione, 99 volte su 100 va in crash, resettando il computer.

Lo Spectrum del QL

Cara redazione di SUPERSINC, sto per comprarmi uno Spectrum, ma prima vorrei essere sicuro di una cosa: mi domando se dopo l'uscita del QL la produzione dello Spectrum diminuirà o se continuerà ad essere venduto normalmente.

Simone Catani - (Ascoli Piceno)

Caro Simone,
non ti preoccupare: il QL e lo Spectrum appartengono a fasce di mercato estremamente diverse fra di loro, e di conseguenza non si "pesteranno i piedi". E la Sinclair non intende mettere da parte lo Spectrum, anzi: non hai letto nello scorso numero la notizia dell'arrivo dello ZX Spectrum + ?

Gestione aziendale

Mi chiedo se uno ZX Spectrum sia in grado (anche con espansioni) di essere utile anche in campo aziendale, e perciò se si trovano programmi adatti.

Leonardo Corallini Civitanova Marche (MC)

Certo. Specie con una tastiera professionale (o magari con lo Spectrum +) è possibile, disponendo dei 48K, almeno un microdrive e una stampante "seria", lanciarsi nel campo gestionale. I programmi di questo tipo sono numerosissimi, e sinceramente non sapremmo cosa consigliare. Per quanto riguarda i libri, invece, le raccomandiamo "La gestione delle informazioni con lo ZX Spectrum", edito dalla McGraw-Hill e già recensito su SUPERSINC.

VIC o Spectrum?

Caro SUPERSINC,
in questi giorni mi comprerò un computer, ma sono indeciso se comprarmi un Commodore VIC 20 o lo Spectrum. Ti prego di rispondermi (consigliandomi saggiamente) al più presto.

Michele Giannarelli Ceparana (SP)

Caro Michele,
scusaci per il ritardo. Senza sbilanciarci troppo, diciamo che il VIC 20 è più economico dello Spectrum e possiede una vastissima mole di programmi, ma offre prestazioni nel complesso inferiori, a cui è possibile porre rimedio con costose espansioni. In più, tieni presente che il VIC 20 è ormai al tramonto, surclassato com'è dalle performance del superiore Commodore 64, mentre lo Spectrum ha ancora davanti a sé un periodo di vita soddisfacente. Contento?

Domande a go-go

Cara redazione di SUPERSINC,
è la prima volta che vi scrivo, anche se vi seguo dal primo numero. Io posseggo uno ZX Spectrum ed avrei alcune domande da farvi:
1) può lo ZX venire espanso fino oltre i 48K? Per mezzo di che cosa?
2) ho sentito parlare di programmi chiamati copiatori, qual è il migliore in commercio?
3) qual è il miglior joystick per lo ZX?
4) per installare il sintetizzatore vocale bisogna fare qualche modifica nei circuiti dello ZX o aggiungere qualche interfaccia?

Marco Gorin - Vercelli

Caro Marco,
andiamo con ordine: 1) la memoria dello Spectrum può essere portata senza aggiunte esterne fino ad 80K, utilizzando lo spazio apposito che esiste in alcuni Spectrum (aprite il calcolatore: se non avete spazio, saltate alla prossima lettera) in cui è possibile inserire chip di RAM uguali o equivalenti a quelli già presenti. In ogni caso gli 80K non sono mai disponibili tutti insieme, e possono essere sfruttati in modo ottimale solo con programmi che prevedano la cosiddetta "gestione a banchi"; 2) non ci sentiamo, a questo proposito, di dire alcunché; 3) non vogliamo fare torto a nessuno! Diciamo solamente che il joystick più diffuso è quasi sicuramente il Kempston; 4) dipende da che tipo di sintetizzatore vocale vuoi installare! Quelli commerciali, in genere, si interfacciano allo Spectrum tramite il connettore posteriore, dovresti essere più preciso...

di **Marcello Spero**

Spectrum e interfaccia 1: un sistema completo

Quali sono le possibilità d'uso dell'interfaccia 1?

Dopo avere esaminato i microdrive, sia dal punto di vista del funzionamento che da quello delle applicazioni pratiche, questa volta ci occuperemo delle altre possibilità offerte dall'interfaccia 1 che, nota soprattutto per essere "l'interfaccia dei microdrive", offre all'utilizzatore altre notevoli possibilità.
Lo Spectrum in "configurazione estesa", cioè con l'interfaccia collegata, viene infatti a disporre di due canali di collegamento con l'esterno: la rete locale e l'RS232. In più, in questa condizione si ha una modificazione sostanziale nella struttura dell'interprete BASIC, che consente l'aggiunta di nuovi comandi creati dall'utilizzatore. Ma procediamo con ordine.

La rete locale

Consente il collegamento di un numero notevole di Spectrum, normalmente fino a 64, in modo da rendere possibili comunicazioni a velocità elevata. L'aggettivo locale sottolinea il fatto che questo tipo di collegamento è concepito per brevi distanze.
La sua struttura è semplicissima: ogni Spectrum è collegato ad altri due per mezzo delle due prese, simili a quelle per il registratore, poste sul retro dell'interfaccia. Da ogni presa parte un cavo che va ad inserirsi nell'identica presa di un altro computer, formando una catena attraverso la quale scorrono le informazioni.
La velocità di trasferimento di queste informazioni è, come dicevamo, molto elevata, dell'ordine di 5 Kbyte per secondo. Le informazioni possono essere inviate ad uno solo dei computer collegati, o a tutti, e lo stesso vale per la ricezione; sono considerate dei file a tutti gli effetti, e perciò possono essere, come per i microdrive, di tipo "programma" o "dati".
Prima di addentrarci di più nel funzionamento della rete, è opportuno introdurre una terminologia minima, necessaria al nostro discorso.
In ogni rete, e quindi anche in questo caso, ciascun computer che vi partecipa si dice stazione. Le stazioni si distinguono l'una dall'altra per mezzo di un numero, il numero di stazione, appunto. Questi numeri servono a far capire a tutta la rete da chi proviene ed a chi è destinato un determinato messaggio (o file, che dir si voglia).
La stazione che invia un messaggio è la sorgente; quella che lo riceve, la destinazione.
Vediamo adesso come si usa in pratica la rete locale.
All'accensione (piú esattamente dal primo uso dell'interfaccia dopo l'accensione) lo Spectrum assume il numero di stazione 1. Ovviamente in una rete non potrà esserci più di un computer con questo numero, e quindi gli altri dovranno assegnarsene un altro. Questo avviene con il comando

FORMAT "n"; numero di stazione

I numeri di stazione vanno normalmente da 1 a 64.
Una volta che tutti abbiano compiuto questa operazione, la rete è pronta a funzionare. Le modalità di invio dei file sono analoghe a quelle per i microdrive; per un file "programma", infatti, basta il comando

SAVE * "n"; 2

In questo caso il messaggio, un programma, è diretto alla stazione 2. L'invio, comunque, non avverrà finché la stazione interessata alla ricezione (la 2 in questo caso) non si predisporrà con

LOAD *"n"; 1

(supponendo che la sorgente abbia numero di stazione 1). Non appena la 2 avrà dato l'ENTER relativo al comando inizierà a ricevere il programma, che apparirà sul suo schermo; contemporaneamente la 1 vedrà lampeggiare il bordo dello schermo, a segnalare la trasmissione in corso.
Lo stesso vale per le altre forme di file "programma", e cioè i DATA ed i CODE (o SCREEN$ a seconda dei casi). Al comando

SAVE *"n"; 2 DATA a ()

dovrà corrispondere

LOAD *"n"; 1 DATA

Come avrete notato, non viene fatto uso dei consueti nomi, che viceversa con i microdrive sono obbligatori anche in fa-

Fig. 1 - Lo Spectrum collegato all'interfaccia 1.

se di LOAD . In questo caso, infatti, non sarebbero di alcuna utilità, visto che i file non dovranno essere ritrovati in futuro in base al loro nome, ma sono inviati ad un destinatario già pronto a riceverli.
Comandi del tipo

SAVE *"n";2;"pippo"

o

LOAD *"n";1;"prezzi" DATA

sono comunque possibili, anche se i nomi vengono ignorati. A questo proposito, state attenti a non scrivere

LOAD *"n";1; DATA

perché se mettete il punto e virgola dopo il numero di stazione, la macchina si aspetterà un nome, e non trovandolo darà errore.
Per quanto riguarda i programmi, poi, è possibile, in risposta ad un SAVE, dare

VERIFY *"n";1

per verificare il programma già in memoria con quello in ricezione, od anche

MERGE *"n";1

per aggiungere il programma in ricezione a quello già in memoria.
Come abbiamo detto, anche qui esistono i file "dati", per i quali valgono le stesse norme viste con i microdrive.
Supponiamo di essere la stazione 1, che si vuole mettere in contatto con la 10. Per prima cosa dovremo aprire un file con

OPEN #7;"n";10

dove il primo numero è quello del flusso associato al file, cioè quello che indicheremo nelle successive istruzioni di trasmissione o ricezione; il secondo è invece il numero della stazione con cui vogliamo metterci in contatto.
Ovviamente, anche la stazione 10 dovrà fare lo stesso; mentre il numero del flusso potrà essere a sua scelta, quello della stazione dovrà essere l'1, pena l'impossibilità di qualsiasi comunicazione.
A questo punto una delle due stazioni, ad esempio la 1, invierà all'altra dei dati, con una o più

PRINT #7; a'b'c'...

che saranno ricevuti dalla 10 con altrettante

INPUT #4;1;m;n;...

La scelta dei caratteri separatori non è casuale. Perché più dati possano esser riconosciuti come distinti, infatti, devono essere separati da ENTER (CHR$ 13); questo si ottiene usando una PRINT per ciascun dato, o adoperando come separatore il carattere "," (CHR$ 39). Nelle INPUT, viceversa, è necessario usare sempre il carattere "," (CHR$ 59).
Il primo movimento di dati per ogni file ne definisce la direzione. In questo caso, dopo le operazioni descritte, il file della stazione 1, aperto sul flusso 7, sarà di scrittura, mentre quello della stazione 10, aperto sul flusso 4, sarà di lettura. D'ora in avanti ogni tentativo da parte di 1 di ricevere dati su #7, o da parte di 10 di inviarne su #4, daranno errore; l'unico modo per cambiare la direzione di un file è infatti chiuderlo e riaprirlo.
Come per i microdrive, anche qui è possibile l'uso di INKEY$ al posto di INPUT, per leggere un singolo carattere per volta. La stringa "pippo", ad esempio, contenuta in una singola variabile a$, ed inviata mediante un'unica PRINT, potrà essere letta da cinque INKEY$, che ne assegneranno i caratteri ad altrettante variabili; una sesta INKEY$ leggerà il carattere ENTER al termine della stringa.
Questo sistema è valido anche per i numeri; questi, infatti, sono inviati lungo la rete in forma di stringhe di caratteri. Il numero 5 sarà una stringa contenente "5" e ENTER, il numero 123 sarà invece di quattro caratteri (i suoi tre più ENTER) e così via. INKEY$, dunque, potrà scomporre un numero nelle cifre che lo compongono, assegnandole ad altrettante variabili stringa.
INKEY$ ha comunque un'altra caratteristica che ne rende interessante l'uso. Se, infatti, ci poniamo in attesa di dati provenienti, ad esempio, dalla stazione 4, con una INPUT, il bordo dello schermo diventerà scuro ed il computer rimarrà bloccato finché non gli arriverà ciò che aspetta. Usando INKEY$, invece, se la stazione 4 non sta trasmettendo niente nel momento in cui questa viene eseguita, otteniamo una stringa vuota, ed il programma prosegue. Questo consente il cosiddetto "polling", cioè l'esame veloce di molte stazioni, raccogliendo dati da tutte quelle che ne stanno trasmettendo.
La trasmissione di file di questo tipo non è continua; i dati di ciascuna PRINT vengono accumulati in un "buffer", cioè un'area di attesa in memoria, ed inviati solo quando questo è pieno (ogni 255 caratteri) o quando il file viene chiuso con CLOSE#. La trasmissione di ciascun blocco di dati è resa evidente dal lampeggio del bordo dello schermo.
C'è ancora una caratteristica, comune sia ai file "dati" che a quelli "programma", che vale la pena di ricordare. Esiste un canale, detto "di trasmissione collettiva", contrassegnato dallo 0. I file trasmessi usando come numero di stazione lo 0 potranno essere ricevuti da tutte le stazioni che si siano messe in ascolto per la stazione 0. In questa situazione anche INKEY$ attende la trasmissione.
Gli usi di uno strumento flessibile come la rete locale possono essere i più disparati. Si pensi ad una classe, in cui ogni allievo abbia davanti a sé uno Spectrum. L'insegnante, anch'egli provvisto di computer, potrebbe inviare un esercizio a tutta la classe; ogni allievo, a sua volta, potrebbe inviare all'insegnante la propria soluzione, senza che gli altri la possano vedere.
Un'altra interessante applicazione si potrebbe trovare nell'elaborazione parallela di grandi masse di dati, che verrebbero suddivise fra più macchine; i vari risultati, inviati ad un unico computer, verrebbero utilizzati per le successive elaborazioni. Questo sistema consentirebbe un notevole risparmio di tempo.
E cosa dire di una battaglia simulata, combattuta da due Spectrum collegati in rete?

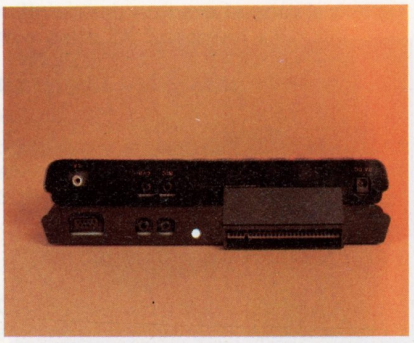
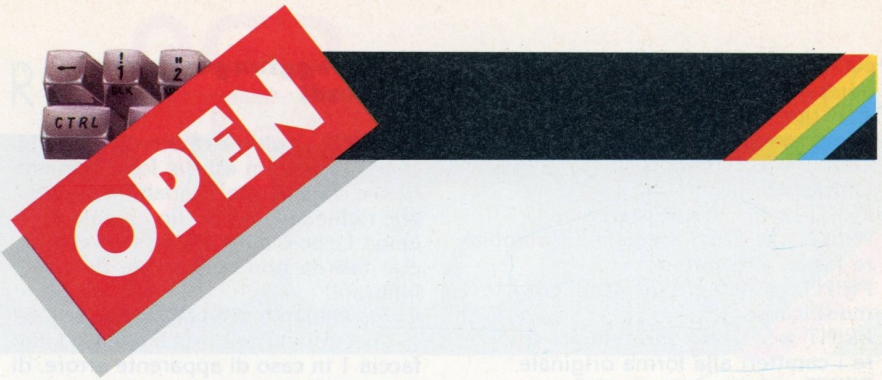


Fig. 2 - Da sinistra a destra, sono visibili sull'interfaccia il connettore di espansione, gli ingressi per il collegamento alla rete, l'interfaccia RS232 e il connettore per microdrive.

L'interfaccia RS232

Si tratta di un'interfaccia di comunicazione di tipo seriale. Questo vuol dire che le informazioni possono essere sia trasmesse che ricevute e si muovono utilizzando un'unica linea, come per la rete locale.
Le sue caratteristiche (ordine in cui vengono trasmessi e ricevuti i vari segnali, numero e tipo dei collegamenti, e così via) sono di tipo standard; sono cioè le stesse per tutte le interfacce, di qualsiasi provenienza, siglate RS232. Sarà perciò possibile il collegamento fra apparati anche molto diversi, purché entrambi possiedano questo tipo di "conversione" dei loro dati nel linguaggio universale della RS232.
Lo Spectrum può in questo modo collegarsi a stampanti non ZX, ad altri computer sia a breve che a lunga distanza, eventualmente per mezzo di un "modem" che permetta l'uso della linea telefonica, o a qualsiasi apparecchiatura o strumento con cui si renda necessario lo scambio di informazioni.
Abbiamo detto che il modo in cui i dati si muovono fra due RS232 è lo stesso utilizzato dalla rete locale. Questo è vero solo in linea generale. L'RS232, infatti, utilizza due linee separate per i due sensi di movimento delle informazioni, più altre due per particolari segnali che servono a "mettere d'accordo" entrambi gli apparecchi collegati.



La velocità con cui le informazioni vengono trasferite è determinata da un particolare parametro, il "baud rate", cioè il numero di bit al secondo.
Poiché un byte è composto da otto bit, ed altri tre sono necessari per le informazioni riguardanti la linea, il numero di byte, cioè di singole unità di dati, trasferiti ogni secondo sarà uguale a

baud rate/11

I baud rate che lo Spectrum può assumere sono

50,110,300,600,1200,2400,4800,9600, 19200

cui corrispondono velocità teoriche da 4.5 a 1745.4 byte al secondo. Come vedete, si tratta di velocità decisamente inferiori a quella della rete locale. Il ritmo reale di trasferimento dei dati è comunque ancora più basso, poiché fra un byte e l'altro sono necessarie delle pause per attendere i segnali di "pronto" sulle due linee accessorie.
La presa di collegamento dedicata all'RS232 nell'interfaccia 1 non è di tipo standard. Per poter effettuare qualsiasi collegamento è perciò necessario acquistare o fabbricarsi un convertitore, che da un lato possieda una spina adatta alla presa sull'interfaccia 1 e dall'altro un connettore standard. A beneficio di quanti volessero provvedere all'autocostruzione di questo semplice dispositivo, l'appendice del manuale di uso riporta lo schema dei collegamenti da effettuare.
Una volta collegato fisicamente, cioè per mezzo di un cavo adatto, lo Spectrum con l'apparecchio (stampante, altro computer, ecc.) con cui si deve svolgere lo scambio di informazioni, occorre scegliere opportunamente il baud rate. La scelta può essere di due tipi. Se, infatti, l'apparecchiatura collegata possiede un solo baud rate, cioè può trasferire dati ad un'unica velocità, non c'è scelta: sarà questa a dover essere scelta anche da noi. Nel caso, invece, che siano possibili più baud rate diversi andrà selezionato, per ambedue gli apparecchi, il più alto compatibile con il tipo di linea usato. Nelle comunicazioni fra computer attraverso la linea telefonica, ad esempio, non avrebbe senso optare per un valore di 19200, in quanto tale valore, anche se possibile per ambedue gli apparecchi, darebbe luogo ad errori, a causa delle limitazioni proprie delle linee telefoniche.
Lo Spectrum esteso risulta selezionato, all'accensione, per il valore di 9600 baud. Per scegliere qualsiasi altro valore basta dare

FORMAT "b";110

dove il valore 110 è quello che viene selezionato.
Da questo momento sono possibili tutte le operazioni viste per la rete locale. I file "programma" saranno inviati con

SAVE *"b"

per programmi,

SAVE*"b"DATA a ()

per dati, e

SAVE*"b" CODE x,y

per i byte, e ricevuti con

LOAD*"b"

LOAD*"b"DATA

LOAD*"b"CODE

Anche qui è possibile l'uso di MERGE e VERIFY, con le stesse modalità. Anche per i file "dati" valgono le stesse regole della rete. Con

OPEN #4;"b"

il flusso 4 viene associato all'RS232. Sono ora possibili le consuete operazioni di trasmissione, con

PRINT #4;a'b'c'...

e ricezione, con

INPUT #4;a;b;c;...

che attende una trasmissione, o

LET a$=INKEY$ #4

che riporta un solo carattere, o la stringa vuota, se non sono in corso trasmissioni.
Qui, a differenza che con la rete, è possibile utilizzare uno stesso flusso sia in trasmissione che in ricezione, poiché non vengono creati buffer di accumulo, ma i caratteri sono inviati uno per volta. La serie di istruzioni

OPEN #4;"b"
PRINT #4;a$
INPUT #4;b$

è perciò consentita.
Quando, terminato l'utilizzo di un flusso, lo si chiude con

CLOSE #4

viene inviato sulla linea un carattere di "andata a capo" (CHR$10). Questa caratteristica può essere utile con le stampanti, ma in altre situazioni può risultare di impaccio. In questi casi è possibile chiudere un flusso con

CLEAR #

che non invia alcun carattere in più.
Per i file di tipo "dati" esiste anche un altro sistema di trasmissione: il sistema "t". Un flusso associato a questo sistema con

OPEN #4;"t"

avrà queste caratteristiche:
- i caratteri dallo 0 al 31 vengono ignorati, eccetto il 13 (ENTER) cui è aggiunto un CHR$ 10 (LINE FEED, andata a capo);
- i caratteri grafici sono trasformati in punti interrogativi (CHR$ 63);
- le parole del BASIC non vengono trasmesse come singoli codici, ma sono espanse nei caratteri che ne compongono il nome. LET, ad esempio, non sarà inviato come CHR$ 241, ma come CHR$ 32 + CHR$ 76 + CHR$ 69 + CHR$ 84 + CHR$ 32 (il primo e l'ultimo carattere sono degli spazi).
Questo modo di trasmettere è stato concepito espressamente per pilotare correttamente le stampanti commerciali nel caso di un invio di listati, che devono essere "ripuliti" da tutti i caratteri non standard usati dallo Spectrum. Resta comunque possibile l'invio di caratteri spe-

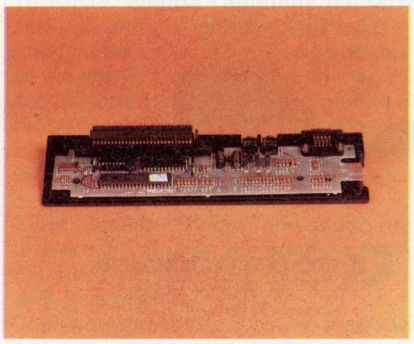


Fig. 3 - Visione interna dell'interfaccia.

ciali di controllo ad una stampante, aprendo semplicemente anche un flusso di tipo "b", che sarà usato solo in particolari occasioni:

```
OPEN #4;"t"
OPEN #5;"b"
PRINT #4;"testo da stampare"
PRINT #5; CHR$ speciale per cambiare forma ai caratteri
PRINT #4;"scritta con caratteri modificati"
PRINT #5; CHR$ speciale per riportare i caratteri alla forma originale
PRINT #4; "testo normale"
```

proteggendo così la stampa dai caratteri non standard.
Per finire un accenno al comando MOVE, che in unione a rete e RS232 acquista possibilità notevoli. Con

```
MOVE "n"; 10 TO "t"
```

si ottiene l'invio di tutte le informazioni, inviate attraverso la rete dalla stazione 10, ad una stampante collegata allo Spectrum ricevente.

```
MOVE "b" TO "m";1;"pippo"
```

invece, consente l'uso dei microdrive da parte di un computer di altra marca, collegato allo Spectrum attraverso l'RS232. Notevole, vero?

Come aggiungere nuovi comandi

Ed eccoci all'ultima caratteristica del sistema Spectrum esteso, forse la meno nota e considerata, probabilmente perché richiede l'uso del linguaggio macchina. Le possibilità offerte sono comunque tali da non consentirle di essere ignorata.
Come abbiamo visto la scorsa puntata, lo Spectrum inserisce la ROM dell'interfaccia 1 in caso di apparente errore, di sintassi o di funzionamento. In questi casi viene concessa al comando incriminato "una prova di appello"; lo Spectrum verifica cioè se il presunto errore non sia in realtà uno dei nuovi comandi, aggiunti dalla ROM dell'interfaccia. In caso affermativo questa eseguirà tutte le operazioni necessarie, eventualmente emettendo uno dei suoi messaggi di errore; altrimenti viene effettuato un ritorno alla ROM principale.
Questo ritorno avviene, in condizioni normali, all'indirizzo corrispondente alla routine di errore. La vera novità sta comunque nel fatto che questo indirizzo di ritorno non è contenuto in ROM, e quindi fisso ed inalterabile, ma in RAM, e precisamente in una delle nuove variabili di sistema introdotte dall'interfaccia 1: VECTOR.
Di conseguenza, ponendo in VECTOR l'indirizzo di una routine creata da noi, in RAM, daremo ai comandi risultati sbagliati sia per la ROM principale che per quella dell'interfaccia un'ulteriore possibilità. La nostra routine dovrà:
- esaminare il comando e decidere se si tratta di uno di quelli "extra" da noi creati;
- in caso affermativo, controllarne la sintassi, eventualmente emettendo l'appropriato (nuovo) messaggio di errore, ed eseguirlo, se è il momento giusto;
- altrimenti tornare alla routine di errore originale.

La creazione di una routine di questo tipo non è certamente un'operazione semplicissima, ma il risultato può compensare la fatica. Comunque, per coloro che non se la sentissero di avventurarsi in questo tipo di impresa, non è esclusa la pubblicazione di una serie volta a presentare qualche nuovo comando già pronto, sotto forma di routine da trascrivere.
Si conclude qui la nostra chiacchierata sull'interfaccia 1 e le sue possibilità. Ovviamente i singoli argomenti sono stati trattati in modo piuttosto generale; se ci arriveranno le vostre richieste, saremo comunque lieti di esplorare più a fondo i singoli aspetti dell'uso di questa notevole periferica Sinclair.

Verifica che il tuo Commodore 64* abbia gli standard specifici per l'Italia.

* e il Commodore 16

Commodore Italiana desidera aiutarti a scegliere bene il tuo Commodore 64.*
Un buon consiglio:
prima di acquistarlo, chiedi al tuo rivenditore di mostrarti se è un 64 approvato e collaudato per l'Italia.

Perchè è meglio controllarlo.
Perchè quello non collaudato e non originale, cioè non distribuito regolarmente dalla Commodore Italiana, non è nato per i nostri standard elettronici, elettrici, qualitativi, e non è protetto per le interferenze radio.

Come fai a riconoscerlo.
Dal trasformatore/alimentatore: se è un Commodore non collaudato per l'Italia, il trasformatore/alimentatore è previsto per una tensione più alta (240 volt, cioè per lo standard di altri Paesi d'Europa); se è un Commodore collaudato e perciò garantito, la tensione è quella giusta, 220 volt.

Scegli bene.
Solo comperando un Commodore 64 originale, la Commodore Italiana potrà garantirti che il tuo grande personal funzionerà al meglio della sua potenzialità.

commodore COMPUTER

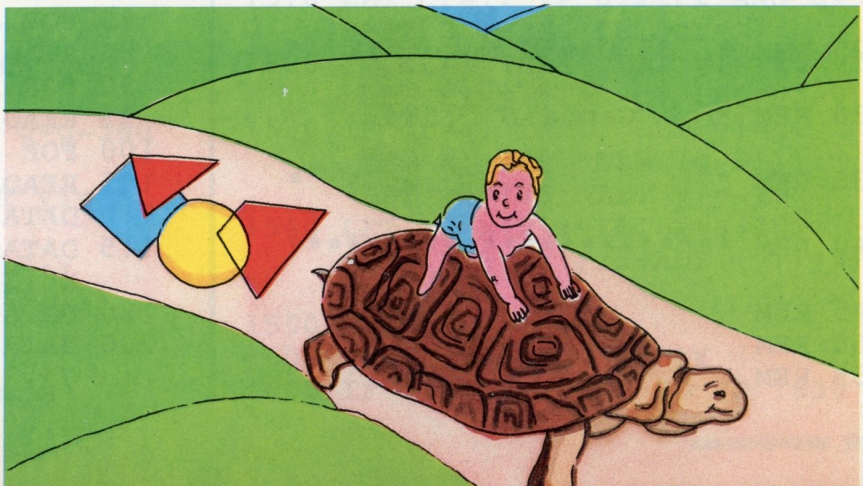
Logo e tartarughe

di **Tim Hartnell**
trad. e adatt.
di **Marcello Spero**

Seconda parte

Un microinterprete Logo scritto in BASIC

Nel numero scorso abbiamo presentato il linguaggio di programmazione Logo.
Senz'altro molti di voi vorranno ora provare Logo sul loro Spectrum, ma la ricerca di software commerciale in questo campo è davvero scoraggiante: sul mercato non c'è quasi niente.
Per rimediare a questa situazione presentiamo, in una serie di articoli, un programma BASIC che simula il Logo. Chi avesse perso l'articolo introduttivo non si preoccupi: avremo occasione di riprenderne i concetti fondamentali.
In ciascuna delle parti che compongono questa serie troverete un programma BASIC da utilizzare con lo Spectrum ed istruzioni sul modo di usarlo.
Usando questi programmi, in effetti, programmerete in Logo. Alla fine della serie, quindi, vi ritroverete a possedere una discreta padronanza di questo linguaggio.
Il programma della prossima parte andrà unito a quello che trovate in questo articolo, facendo uso dell'istruzione MERGE; il terzo programma della terza puntata, infine, andrà aggiunto ai primi due allo stesso modo.
Fino al secondo programma potranno arrivare anche i possessori di Spectrum 16 Kbyte, mentre l'aggiunta del terzo sarà alla portata dei soli 48 Kbyte.
Idealmente, un interprete Logo dovrebbe essere scritto in linguaggio macchina, per ottenere un'elevata velocità di esecuzione. Questa versione è invece in BASIC, che ha reso più facili le cose nello scriverla e le renderà più facili a voi nel copiarla e comprenderla.
Certo, usando il BASIC non si può pretendere la velocità, ma anche a passo di tartaruga questo programma può essere interessante.
Una volta introdotto il listato e salvato il programma su cassetta sarete pronti per iniziare. Prima di provare a dare i nostri primi comandi in Logo, però, può essere utile dare una scorsa ai principi fondamentali di questo linguaggio.
Logo è stato progettato per dare ai bambini una precoce istruzione alla programmazione dei computer ed aiutarli a sviluppare il pensiero logico. I comandi del Logo sono semplici da comprendere e la sua particolare struttura incoraggia una programmazione corretta.
Il Logo è più noto come linguaggio grafico, dato che rende possibile la creazione di figure immobili od animate, nonché la generazione di motivi geometrici. Le versioni più potenti del Logo possiedono particolari possibilità anche in altri campi; questa, come la maggior parte, è unicamente grafica.
Troverete il Logo estremamente utile, se avete a che fare con bambini, ma anche se siete in cerca di una versatile routine grafica. Se volete staccarvi dal BASIC, inoltre, il Logo rappresenta un mezzo che, pur semplice da imparare, vi introdurrà nel mondo della programmazione strutturata, usata da linguaggi più potenti come il Forth.
La grafica del Logo fa uso di una tartaruga; prima di protestare con la Protezione Animali, comunque, sappiate che si tratta di un animale immaginario!
Supponete di avere un completo controllo sui movimenti della vostra tartaruga e di poterla quindi istruire a muoversi in avanti o indietro, a destra od a sinistra, di una lunghezza da voi stabilita. La vostra tartaruga, oltretutto, è munita di una penna che può sollevare od abbassare al vostro comando. Quando la penna è abbassata, la tartaruga lascerà una traccia del suo passaggio, disegnando una linea.
Con la combinazione dei quattro movimenti (avanti, indietro, sinistra, destra) si possono creare disegni e motivi geometrici.
Alcune versioni del Logo si servono si autentiche tartarughe meccaniche, interfacciate ad un computer. I vari comandi Logo sono trasmessi alla tartaruga, che traccia delle forme su di un foglio di carta posto sul pavimento. Più spesso è il monitor o il televisore a fare da foglio di carta, e quella che viene mossa è una tartaruga elettronica. È così che funziona questa versione.
Fate partire il programma e vedrete sullo schermo la nostra tartaruga. È rappresentata da un "~"; la punta ne indica la testa, cioè l'attuale direzione. Vedrete che il punto di partenza della tartaruga, rivolta verso l'alto, è al centro dello schermo.
Nella parte bassa dello schermo vedrete invece una "W;" ed un cursore lampeggiante. Ciò sta ad indicare che il computer è in attesa di un comando.
In questa prima fase il programma accetta solo gli otto comandi che vedete in figura 1. Notate come tutti i comandi vadano scritti in lettere maiuscole (CAPS LOCK è inserito automaticamente dal programma) e come per alcuni di



essi sia possibile usare un'abbreviazione di due lettere al posto del comando completo.
Prima che iniziate ad usarli, vediamo come agisce ognuno di essi.
CENTRO: muove la tartaruga da un punto qualsiasi dello schermo alla sua posizione di partenza, al centro e rivolta in su.
PRONTI: pulisce lo schermo e porta la tartaruga al punto di partenza.
AVANTI: la tartaruga si muove in avanti del numero di passi da voi stabilito. Per completare il comando dovrete infatti dare questo numero; es.: AVANTI 20 (o AA 20). L'unità di misura della distanza è un pixel. Lo schermo, ricordatelo, è largo 256 pixel ed alto 176.
INDIETRO: la tartaruga fa dietrofront e quindi si muove nello stesso modo descritto per AVANTI. Es.: INDIETRO 25 (o II 25).
SINISTRA: la testa della tartaruga rimane nel medesimo posto ed il suo corpo le ruota intorno, in modo da girarla verso sinistra. È necessario dirle di quanto deve ruotare. Come AVANTI e INDIETRO, infatti, anche SINISTRA richiede un numero, in aggiunta al comando, per essere completo. Questo numero rappresenta l'angolo della rotazione, espresso in gradi. Se avete dubbi, potete usare la figura 2 per aiutarvi con i comandi di rotazione. Gli angoli dovranno essere numeri compresi fra 0 e 360.
DESTRA: segue gli stessi principi di SINISTRA, naturalmente ruotando la tartaruga in senso opposto.
PENNASU: questo comando fa sollevare la penna dal foglio, in modo che la tartaruga possa muoversi senza lasciare traccia.
PENNAGIU: la penna viene abbassata sul foglio.
Proviamo adesso con qualche esempio. Scrivete:
AVANTI 40 (o AA 40)
e premete ENTER. Vedrete la tartaruga scomparire, sarà tracciata una linea lunga 40 pixel e la tartaruga riapparirà. Per rendere il programma il più veloce possibile la nostra tartaruga scomparirà sempre quando è in movimento; riapparirà quando tutte le istruzioni saranno state eseguite e comparirà il simbolo "W;". Notate anche come alla partenza del programma la penna sia giù, per permetterci di tracciare una linea.
Ruotiamo ora la tartaruga di 90 gradi a destra. Scrivete:
DESTRA 90 (o DX 90)
e premete ENTER.
Ora, per muoverla nuovamente in avanti:
AA 40
Continuando a dare alternativamente DX 90 e AA 40, otterrete un quadrato. Scrivere un comando per volta è fastidioso, perciò il Logo consente di unire più comandi. Vediamo come. Per prima cosa puliamo lo schermo e riportiamo in centro la tartaruga, usando il comando PRONTI; quindi scriviamo:
AA 40 DX 90 AA 40 DX 90 AA 40DX90 AA 40 DX 90
ed ENTER.
Lasciate un singolo spazio fra ogni comando ed il suo numero.
L'effetto è di nuovo un quadrato. Vi sarete accorti che abbiamo ripetuto gli stessi due comandi per quattro volte. Fortunatamente, Logo consente la semplificazione delle ripetizioni; questa è una delle cose che vedremo la prossima volta.
Cosa succede se la tartaruga esce dallo schermo? Proviamo a vedere. Puliamo lo schermo con PRONTI e scriviamo:
DX 10 AA 2000
La tartaruga si volterà leggermente sulla destra e quindi si muoverà in avanti di 2000 pixel. Nel momento in cui scomparirà dallo schermo riapparirà dalla parte opposta. Questa caratteristica può essere utile per creare effetti particolari e ci evita il messaggio di errore se usciamo accidentalmente dallo schermo.
Un'altra caratteristica di questa versione del Logo è che i numeri usati con AVANTI, INDIETRO, DESTRA e SINISTRA possono essere sostituiti da numeri casuali. Come esempio, provate:
AA RANDOM 50
La tartaruga si muoverà in avanti di un valore compreso fra 0 e 50.
Può essere abbastanza, per questa volta. Provate a creare disegni, motivi o forme. Il programma possiede una routine di riconoscimento degli errori che vi fornirà, nel caso, un messaggio comprensibile e chiaro.
Nella prossima parte faremo compiere al nostro interprete Logo un balzo in avanti. Arrivederci!

```
 5 REM ************************
10 REM *                      *
15 REM *        LOGO          *
20 REM *                      *
25 REM *  di David Nowotnik   *
30 REM *    Gennaio  1984     *
35 REM *                      *
40 REM ************************
45 REM
50 REM
55 REM
60 READ m: READ n: READ o
65 DIM x$(m,2): DIM w$(n,12):
   DIM f$(o,12)
70 DIM u(m): DIM v(n): DIM g(o
   )
75 LET pp=1: LET wr=1: LET tar
   t=1
80 FOR i=1 TO m: READ x$(i)
85 READ u(i): NEXT i
90 FOR i=1 TO n: READ w$(i)
95 READ v(i): NEXT i
100 FOR i=1 TO o: READ f$(i)
105 READ g(i): NEXT i
110 DATA 6,8,1
115 DATA "AA",3000,"II",3200,"S
    X",3400,"DX",3600
120 DATA "PS",4000,"PG",4050
200 DATA "AVANTI",3000,"INDIETR
    O",3200,"SINISTRA",3400
```

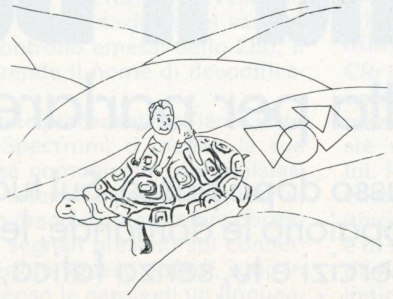
```
205 DATA "DESTRA",3600,"PRONTI"
    ,3800,"CENTRO",3850
210 DATA "PENNASU",4000,"PENNAG
    IU",4050
300 DATA "RANDOM",8000
400 POKE 23658,8
480 GO SUB 3800: GO TO 2000
490 REM
500 REM ERRORI
510 REM
520 GO SUB (690+err*10)
530 PRINT #1;a$
540 PAUSE 250
550 RETURN
700 LET a$="Comando sbagliato"+
    CHR$ 13+"ripeti la linea":
     RETURN
710 LET a$="Numero sbagliato"+
    CHR$ 13+"ripeti la linea":
     RETURN
720 LET a$="La linea"+CHR$ 13+"
    non puo' essere tracciata":
     RETURN
990 REM
1000 REM Subroutine
1010 REM
1020 REM Controllo sintassi
1030 LET t=0: LET y$=""
1040 LET s=s+1: IF s>LEN z$
     THEN  LET t1=1: RETURN
1050 IF (z$(s)=" " OR z$(s)=
     CHR$ 0) AND t=0 THEN
      GO TO 1040
1060 IF (z$(s)=" " OR z$(s)=
     CHR$ 0) AND t=1 THEN
      RETURN
1070 LET y$=y$+z$(s): LET t=1:
      GO TO 1040
1080 REM
1100 REM tartaruga
1110 IF x<3 OR x>252 OR y>172
      THEN  RETURN
1120 FOR j=160 TO 200 STEP 40
1130 LET q=dir+j: IF q<0 THEN
      LET q=360-j
1140 IF q>360 THEN  LET q=q-360
1150 LET q=q*PI/180
1160 LET X1=5*SIN q: LET y1=5*
     COS q
1170 PLOT  INVERSE tart;x,y
1180 DRAW  INVERSE tart;x1,y1
1190 NEXT j: RETURN
1200 REM
1210 REM Numero o funzione
```



```
1220 GO SUB 1020: IF err=1
      THEN  LET err=2: RETURN
1230 IF CODE y$>57 THEN  GO TO 1
     300
1240 IF y$="" THEN  LET err=2:
      RETURN
1250 FOR k=1 TO LEN y$
1260 IF CODE y$(k)<48 OR CODE y$
     (k)>57 THEN  LET err=2
1270 NEXT k
1280 IF err=0 THEN  LET a=VAL y$
1290 LET t=0: RETURN
1300  REM Calcolo funzione
1310 LET y$=(y$+"                ")(
      TO 12)
1320 FOR i=1 TO o: IF y$=f$(i)
      THEN  GO TO 1340
1330 NEXT i: LET err=2: RETURN
1340 GO SUB g(i): RETURN
1400 REM Ritorno X
1410 LET cr=1: IF q<2*PI AND q>
     PI THEN  LET cr=-1
1420 IF cr=1 THEN  LET x1=255-x:
      IF q>1.57 THEN  LET y1=0:
      LET x2=0: GO TO 1430
1425 IF cr=1 THEN  LET y1=x1/
     TAN q: LET x2=0
1430 IF cr=-1 THEN  LET x1=-x:
      LET y1=(x1/TAN (q-PI)):
      LET x2=255
1440 LET y2=y+y1: LET a=a-INT
     SQR (ABS (x1)↑2+ABS (y1)↑2)
1450 RETURN
1470 REM
1500 REM Ritorno Y
1510 LET cr=1: IF q>PI/2 AND q<
     3*PI/2 THEN  LET cr=-1
1520 IF cr=1 THEN  LET y1=175-y:
      LET x1=y1*TAN q: LET y2=0
1530 IF cr=-1 THEN  LET y1=y:
      LET x1=-(y1*TAN (q-PI)):
      LET y2=175: LET y1=-y
1540 LET x2=x+x1: LET a=a-INT
     SQR (ABS (x1)↑2+ABS (y1)↑2)
1550 RETURN
1600 REM Ritorno X e/o Y
1605 IF q>PI/2 THEN  GO TO 1630
```

```
1610 LET x3=x+(175-y)*TAN q
1615 IF x3>255 THEN  GO TO 1400
1620 IF x3<255 THEN  GO TO 1500
1625 GO TO 1680
1630 IF q>PI THEN  GO TO 1645
1635 LET x3=x+y*TAN (PI-q)
1640 GO TO 1615
1645 IF q>3*PI/2 THEN  GO TO 16
     70
1650 LET x3=x-y*TAN (q-PI)
1655 IF x3<0 THEN  GO TO 1400
1660 IF x3>0 THEN  GO TO 1500
1665 GO TO 1680
1670 LET x3=x-(175-y)*TAN (2*
     PI-q)
1675 GO TO 1655
1680 IF y1>=x1 THEN  LET a1=
     INT (y1/COS q+1.5)
1685 IF y1<x1 THEN  LET a1=INT (
     x1/SIN q+1.5)
1690 LET a=a-a1: GO SUB 1450:
      IF y2<0 THEN  LET y2=175
1695 IF y2<0 THEN  LET y2=0
1700 GO SUB 1500: IF x2<0 THEN
      LET x2=255
1705 IF x2>255 THEN  LET x2=0
1710 RETURN
1990 REM
2000 REM Routine di input
2010 REM
2020 INPUT "W:"; LINE z$: LET s=
     0
2025 LET cont=0: LET rc=0:
      GO SUB 2040: GO TO 2020
2030 REM
2040 REM Controllo sintassi
2050 REM
2060 LET t=0: LET t1=0: LET err=
     0
2070 GO SUB 1020
2080 IF t1=1 AND y$="" AND cont=
     0 THEN  LET tart=0:
      GO SUB 1100
2085 IF t1=1 AND y$="" THEN
      RETURN
2090 IF LEN y$<>2 THEN
      GO TO 2130
2100 FOR i=1 TO m
2110 IF y$=x$(i) THEN  GO TO 220
     0
2120 NEXT i
2130 LET y$=(y$+"        ")(
      TO 12)
2140 FOR i=1 TO n
2150 IF y$=w$(i) THEN  GO TO 221
     0
2160 NEXT i
2170 REM Errore di comando
2180 LET err=1: IF tart=1 THEN
      LET tart=0: GO SUB 1100
2190 GO TO 520
2200 GO SUB u(i): GO TO 2220
2210 GO SUB v(i)
2220 IF err>0 THEN  GO TO 520
2230 GO TO 2070
2990 REM
3000 REM Routine AVANTI
3010 REM
3020 GO SUB 1200: IF err>0
      THEN  RETURN
3030 IF tart=0 THEN  LET tart=1:
      GO SUB 1100
3040 LET q=dir*PI/180
3050 LET x1=INT (.5+a*SIN q):
      LET y1=INT (.5+a*COS q)
3060 LET tr=0: LET x2=x+x1:
      LET y2=y+y1
3070 IF x2<0 OR x2>255 THEN
      LET tr=1
3080 IF y2<2 OR y2>175 THEN
      LET tr=tr+2
3090 IF tr>0 AND wr=0 THEN
      LET err=3: RETURN
3100 IF tr=0 THEN  GO TO 3120
3110 GO SUB (1300+tr*100)
3120 IF pp=1 THEN  PLOT x,Y:
      DRAW x1,y1
3130 IF pp=0 THEN  PLOT
      INVERSE 1; OVER 1;x,y:
      DRAW  INVERSE 1; OVER 1;x1
      ,y1
3140 LET x=x2: LET y=y2: IF tr>0
      THEN  GO TO 3050
3150 LET t=0: RETURN
3190 REM
3200 REM  INDIETRO
3210 REM
3220 IF tart=0 THEN  LET tart=1:
      GO SUB 1100
3230 LET dir=dir+180: IF dir>360
      THEN  LET dir=dir-360
3240 GO TO 3020
3390 REM
3400 REM  SINISTRA
3410 REM
```

```
3420 IF tart=0 THEN  LET tart=1:
      GO SUB 1100
3430 GO SUB 1200: IF err>0
      THEN  RETURN
3440 LET dir=dir-a
3450 IF dir<0 THEN  LET dir=360+
     dir: GO TO 3450
3460 RETURN
3590 REM
3600 REM  DESTRA
3610 REM
3620 IF tart=0 THEN  LET tart=1:
      GO SUB 1100
3630 GO SUB 1200: IF err>0
      THEN  RETURN
3640 LET dir=dir+a
3650 IF dir>360 THEN  LET dir=di
     r-360: GO TO 3650
3660 RETURN
3790 REM
3800 REM  PRONTI
3810 REM
3820 CLS
3840 REM
3850 REM  CENTRO
3860 REM
3870 IF tart=0 THEN  LET tart=1:
      GO SUB 1100
3880 LET x=128: LET y=88
3890 LET dir=0: LET tart=0
3900 GO SUB 1100: RETURN
3990 REM
4000 REM  PENNASU
4010 REM
4020 LET pp=0: RETURN
4030 REM
4050 REM  PENNAGIU
4060 REM
4070 LET pp=1: RETURN
4080 REM
8000 REM  RANDOM
8010 GO SUB 1210: IF err=2
      THEN  RETURN
8020 LET a=INT (RND*a): RETURN
```



Per ricevere la cassetta contenente i programmi di SUPERSINC compilate e spedite il coupon qui sotto!

SUPERSINC

CEDOLA DI ORDINAZIONE - CASSETTE
da compilare e spedire in busta chiusa a
J.*soft* - Via Rosellini, 12 - 20124 Milano - Tel. 6888228

Vogliate inviarmi n° ..... cassette
contenenti programmi pubblicati sul numero..... di SUPERSINC al prezzo speciale di L. 15.000 + 2.000 per spese di spedizione cadauna

☐ Contanti allegati ☐ Assegno allegato n° .....

.....

☐ Ho spedito l'importo a mezzo vaglia postale

☐ Ho versato l'importo sul ccp. n° 19445204 intestato a J.*soft* - Milano

☐ Pagherò in contrassegno al postino al ricevimento delle cassette

buono d'ordine per la cassetta contenente i programmi pubblicati sul numero... di SUPERSINC

Nome .....

Cognome .....

Via .....

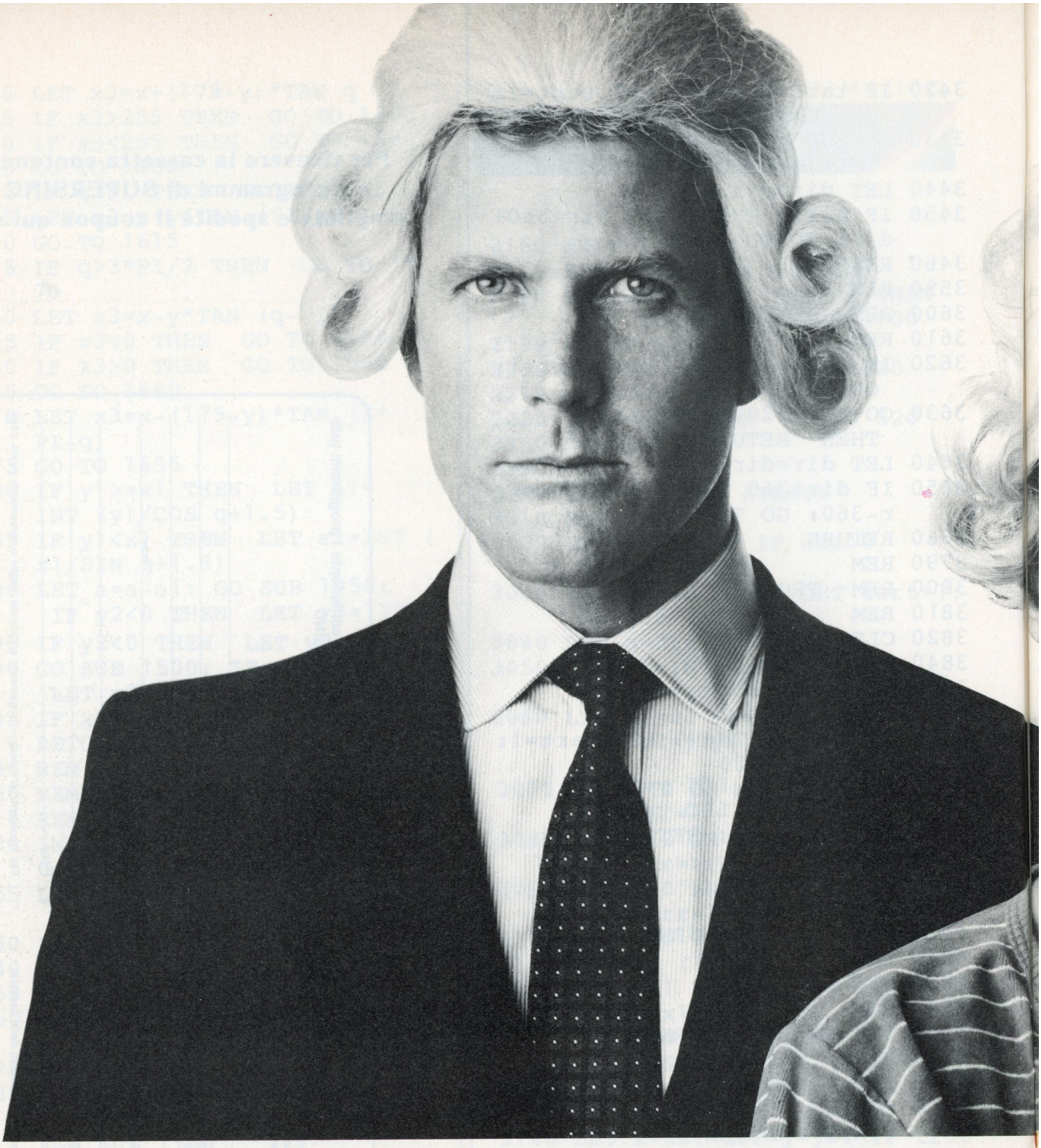
Città ..... C.A.P ..... Prov. .....

Se richiesta fattura:
Cod. F. e P. Iva .....

Data .....

Firma .....

Per i residenti all'estero — pagamento anticipato (vaglia o versamento su ns. ccp)



Il Jacksoniano ha il Basic

Video Basic, corso su cassetta per parlare subito

Oggi è davvero facile imparare il Basic, con Video Basic il corso su cassetta che ti permette di programmare subito il tuo computer. È facile: tu chiedi, lui risponde, tu impari.

Passo dopo passo. Sul tuo schermo appaiono le domande, le risposte, gli esercizi e tu, senza fatica, presto e bene, impari a dialogare col tuo computer, sia un VIC 20, un

REM:HW

Hardware

di **Marcello Spero**

Come realizzare un'interfaccia

seconda parte

La realizzazione pratica di un dispositivo di collegamento

Ed eccoci finalmente all'interfaccia vera e propria o, meglio al decodificatore.
Interfaccia, infatti, è più che altro un termine teorico per indicare quel complesso di apparati atti a consentire lo scambio di informazioni fra due congegni, neanche necessariamente elettronici, che utilizzano codici di comunicazione diversi. Il circuito di cui ci occupiamo oggi, pur rispondendo a questa definizione alquanto generale, ha il compito più specifico di far arrivare alla periferica collegata solo i segnali effettivamente diretti a lei, scegliendoli fra tutti quelli continuamente presenti sul bus dati della CPU dello Spectrum. Esso, inoltre, conserverà ogni segnale fino all'arrivo dell'eventuale successivo, in modo da consentire alla periferica, ben più lenta rispetto allo Z80, di avere il tempo per leggerlo. Infine, ogni volta che la periferica vorrà inviare un segnale allo Spectrum il nostro circuito lo immetterà sul bus dati al momento e nel modo giusto, rendendolo così correttamente interpretabile dalla CPU.
Poiché queste operazioni avvengono mediante l'interpretazione dei vari segnali di controllo emessi dallo Z80, il circuito prende il nome di decodificatore.
Nelle puntate precedenti abbiamo visto come lo Spectrum, o meglio la sua CPU, possa comunicare con qualsiasi congegno esterno, a patto che quest'ultimo sia in grado di interpretare correttamente i segnali presenti sul connettore posteriore. Questi, infatti, sono in un certo senso le parole di un linguaggio particolare, adatto a stabilire un colloquio con altre macchine, allo stesso modo di quanto avviene con gli essere umani per mezzo dei linguaggi di programmazione. Mentre, però, un linguaggio di programmazione serve a dire al computer cosa deve fare, in questo caso è lo Spectrum che dice al mondo esterno cosa sta facendo e cosa vuole; i congegni a lui collegati, ed in grado di comprenderne il linguaggio, potranno solo comportarsi di conseguenza.
Possiamo quindi dire che mentre nello scambio di informazioni uomo-computer è l'uomo ad avere "il coltello dalla parte del manico", in quello computer-periferica (è questo il termine per indicare un congegno esterno collegato ad un computer) questo ruolo lo ha il computer (come per tutte le regole, anche qui ci sono le eccezioni, di cui comunque non ci occuperemo).
Le cose che una periferica deve capire sono in effetti ben poche, sostanzialmente due: quando le informazioni presenti sul bus dati sono dirette a lei e quando invece deve esser lei a porre delle informazioni sul bus. In altre parole, bisogna che una periferica sappia interpretare le richieste di INPUT (entrata) e di OUTPUT (uscita) che lo Spectrum le rivolge.
Ciò che ci accingiamo a costruire sarà un congegno che sappia inviare informazioni solo quando gli vengono richieste e riceverne solo quando sono per lui. Per questo dovrà riconoscere la richiesta di colloquio con una periferica (linea $\overline{IORQ}$), quelle di lettura (linea $\overline{RD}$) e di scrittura (linea $\overline{WR}$), nonché il particolare numero, inviato tramite il bus indirizzi, che lo contraddistingue.
In questo modo avremo creato una specie di controllore del traffico fra il computer e qualsiasi cosa stia dall'altra parte (joystick, stampanti, strumenti vari), cioè un decodificatore.
Passiamo ora ad esaminare il circuito vero e proprio.
In figura 1 vediamo lo schema di principio del nostro decodificatore: le 8 linee del bus dati dello Z80 sono collegate a due congegni, uno per l'ingresso e l'altro per l'uscita. Le due linee OKIN ed OKOUT servono ad attivare l'uno o l'altro. Le possibili situazioni saranno:
— nessuna richiesta di input né di output, figura 2;
— richiesta di input, figura 3;
— richiesta di output, figura 4.
Nel primo caso sia OKIN che OKOUT sono inattive e non si ha perciò alcun flusso di dati attraverso i congegni. Come potete vedere dalla figura, infatti, i segnali provenienti dalla periferica eventualmente collegata non transitano attraverso IN, mentre quelli presenti sul bus non vengono ricevuti da OUT, che invece continua a trasmettere all'esterno l'ultimo dato ricevuto, che ha memorizzato.
Negli altri due casi, invece, uno dei due congegni è attivato dalla linea OK corrispondente, e si ha perciò passaggio di dati in una delle due direzioni.
La figura 3, in particolare, evidenzia come al segnale, introdotto sul bus attraverso l'attivazione di IN, venga impedito di influenzare OUT.
In figura 4, invece, vediamo come il blocco del segnale proveniente da IN eviti un'interferenza con il dato, presente sul bus, diretto a OUT, che è pronto a riceverlo. Se, infatti, una linea del bus fosse contemporaneamente tenuta alta dalla CPU e bassa da IN, si avrebbe un cortocircuito.
Per quanto riguarda il circuito reale abbiamo due casi.
Nella configurazione "minima" (ve ne ricordate? È quella che non consente ampliamenti futuri), infatti, i componenti sono proprio due soli circuiti integrati, uno per l'ingresso e l'altro per l'uscita, più un altro integrato che trasforma i segnali di controllo della CPU in OKIN ed OKOUT.
Il progetto completo, invece, prevedendo l'aggiunta futura di ulteriori canali di ingresso ed uscita, a ciascuno dei quali corrisponderanno due nuovi inte-

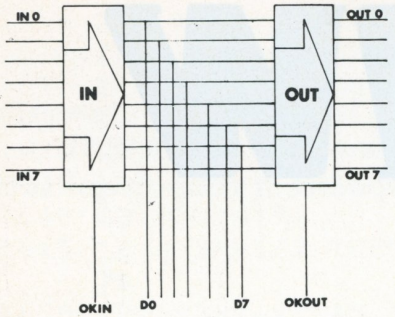


fig. 1 Schema di principio

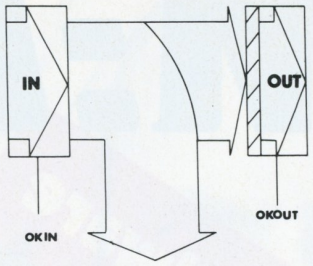


fig. 3 Percorso del segnale in fase di input

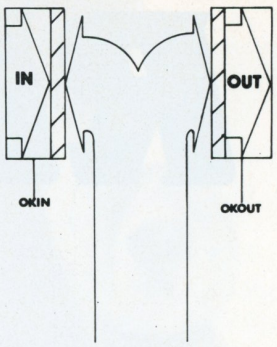


fig. 4 Percorso del segnale in fase di output

grati, deve proteggere il bus dati da un carico eccessivo.
Questa protezione si realizza inserendo un "bus driver". La sua funzione è quella di ricevere i segnali della CPU, che in questo modo alimenta sempre e solo un unico integrato, e ritrasmetterli in avanti con notevole potenza, sobbarcandosi in tal modo l'onere di alimentare tutti i circuiti collegati. Lo stesso avviene per i segnali che si muovono nella direzione opposta e che potrebbero essere stati indeboliti dal numero elevato di circuiti presenti sulla linea.
Le figure 5 e 6 riportano gli schemi completi, rispettivamente per le versioni minima e completa.
Diamo ora un'occhiata ai circuiti integrati utilizzati.

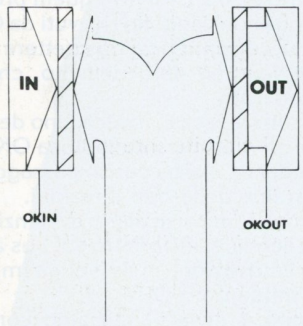


fig. 2 Percorso del segnale in assenza di richieste di input/output

Il controllore di ingresso

Rappresenta in pratica quel congegno che negli schemi teorici abbiamo chiamato IN.
Ciò di cui abbiamo bisogno è, per ciascuna delle 8 linee, una porta logica che trasferisca in uscita il segnale che ha in ingresso solo se riceve un determinato segnale di controllo. Questo tipo di porta si chiama "buffer a tre stati"; in figura 7 ne troviamo la tavola di verità, da cui appare chiaro come in assenza dell'apposito segnale l'uscita non assuma valore alto né basso, ma abbia invece una resistenza elevatissima, che la rende in pratica non collegata all'ingresso (questa particolare situazione rappresenta il "terzo stato" possibile, da cui il nome).
Il circuito integrato che abbiamo scelto, un 74LS244, contiene appunto 8 buffer di questo tipo. Ciascuno di questi, inoltre, è corredato di un particolare dispositivo, il "trigger di Schmitt", che serve ad eliminare la situazione di incertezza che, lo abbiamo visto la volta scorsa, si crea con segnali compresi fra 0.7 e 3 volt, cioè non classificabili né come alti né come bassi. Esso, infatti, considera basso un impulso finché questo non ha superato il valore di 1.7 volt; a questo punto, però, lo stesso impulso verrà considerato alto finché il suo valore non sarà inferiore a 0.7 volt. In questo modo non vengono trasmesse sul bus eventuali fluttuazioni negli impulsi inviati dalla periferica collegata.
La sigla LS (low-power-schottky), infine, sta ad indicare un tipo di integrato a basso consumo, caratteristica che ci

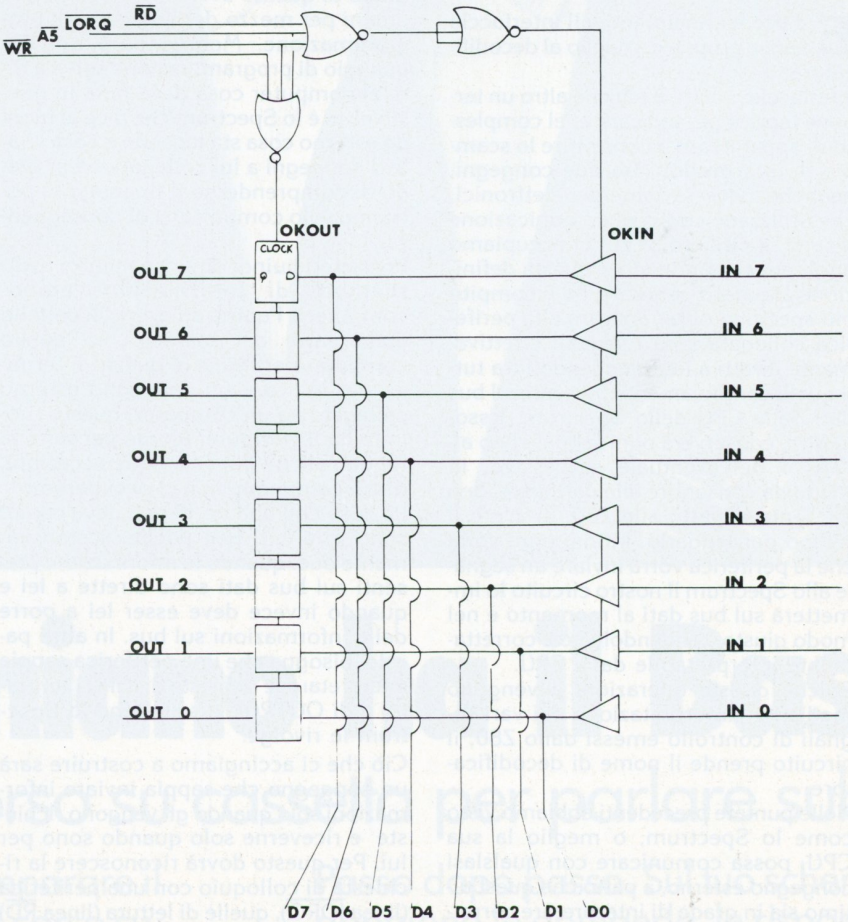


fig. 5 Schema elettrico della versione minima

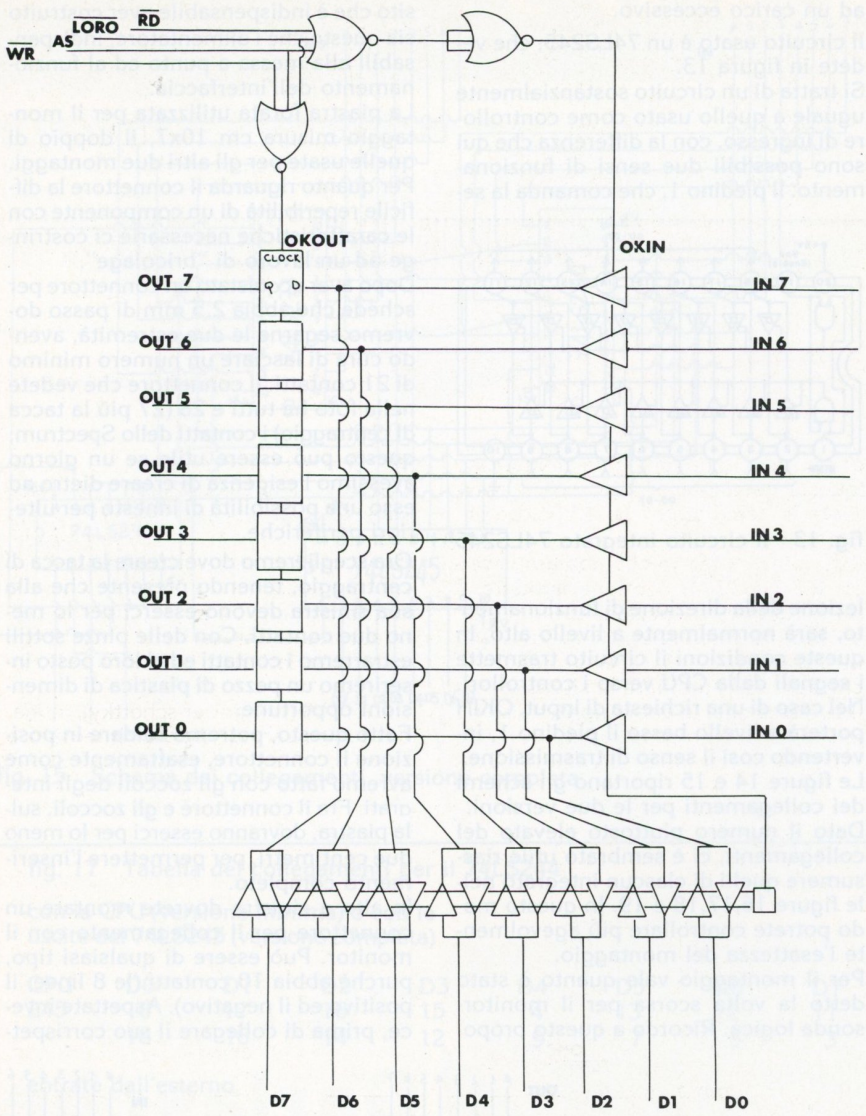


fig. 6 Schema elettrico della versione completa

consente di non utilizzare grossi circuiti di alimentazione; inoltre, per portare a livello basso un ingresso LS sono sufficienti 0.4 mA, contro 1.6 mA di un TTL normale.
La figura 8 ne riporta la struttura interna e la distribuzione delle linee sui vari piedini. I piedini 1 e 19, in particolare, portano le linee di "enable", che a livello basso abilitano il trasferimento in uscita dei segnali in ingresso. Sono due, una ogni 4 linee, che noi collegheremo insieme.

Il controllore di uscita

È il congegno che abbiamo chiamato OUT negli schemi teorici.
Deve, quando è abilitato, trasferire in uscita i segnali che riceve in ingresso. In questo caso, però, essendo la durata del segnale inviato dalla CPU troppo breve perché questo sia comprensibile ad una periferica lenta, occorre che il congegno lo memorizzi, mantenendolo disponibile in uscita fino a quando non viene abilitato a riceverne uno nuovo.
Ciascuna linea farà perciò capo ad un elemento di memoria statica, cioè un "flip-flop". Questo elemento logico, di cui vedete la tavola di verità in figura 9, possiede generalmente 4 linee: un ingresso D, un' uscita Q, un clock ed un clear (talvolta è presente una linea di preset, che noi però non prendiamo in considerazione perché non riguarda la nostra applicazione).
Quando clear è basso, Q è sempre basso, indipendentemente dallo stato di D. Quando clear è alto, invece, il passaggio di clock da un livello basso ad uno alto porta lo stato di D in uscita (Q). Qui sarà mantenuto (memorizzato) fino ad un nuovo passaggio di clock da basso ad alto.
Notate come non sia uno stato di clock ad attivare il meccanismo, ma un passaggio di stato, e come questo sia valido solo in un senso (da basso ad alto) e non nell'altro (da alto a basso).
Il nostro OKOUT, quindi, per attivare il congegno di uscita dovrà passare da 0 a 1 in presenza dell'esatta combinazione di segnali provenienti dalla CPU.

enable	in	out
0	0	0
0	1	1
1	0	*
1	1	*

* = alta resistenza

fig. 7 Tavola di verità di un buffer a tre stati

Il circuito integrato che usiamo, e che vedete in figura 10, contiene 8 flip-flop, tutti comandati da un unico clock. Può essere un 74LS273 o, come nel caso della realizzazione che vedete fotografata, un 74LS374 che, pur avendo la stessa disposizione delle linee sui piedini, dispone di maggior potenza in uscita e della possibilità di disabilitare le uscite (è cioè a tre stati).

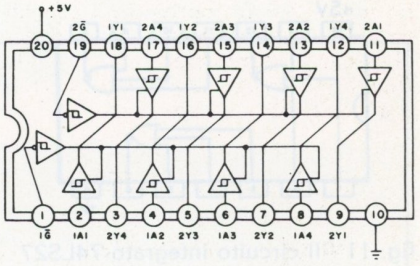


fig. 8 Il circuito integrato 74LS244

Essendoci comunque una piccola variazione nei collegamenti fra un integrato e l'altro, dovuta al fatto che il 273 possiede un terminale di clear, che nel nostro uso va tenuto sempre a livello alto, cioè disinserito, mentre lo stesso piedino nel 374 porta il terminale di "output enable", cioè di abilitazione delle uscite, che va tenuto sempre a livello basso, si è scelto di usare il 74LS273 nello schema della versione minima ed il 74LS374 in quello della versione completa. Nulla vi vieta, comunque, di usare l'integrato meno potente, e meno costoso, anche nell'altra versione.

clear	clock	D	Q
0	1 o 0	1 o 0	0
1	~	0	0
1	~	1	1

~ = passaggio da basso ad alto

fig. 9 Tavola di verità di un flip-flop

La decodifica

Serve a produrre i segnali di OKIN ed OKOUT.
Iniziamo con il precisare che OKIN dovrà essere attivo basso, mentre OKOUT dovrà esserlo alto, o meglio nel passaggio da basso ad alto, poiché così esigono i circuiti usati.

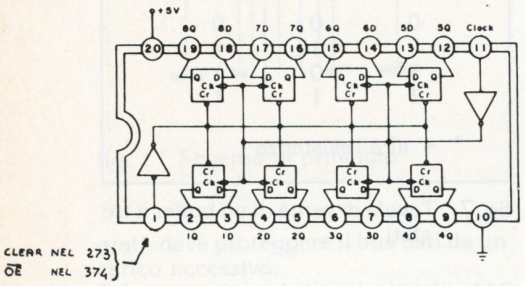


fig. 10 Il circuito integrato 74LS273 e 74LS374

Il circuito utilizzato è un 74LS27, di cui vedete lo schema in figura 11. Si tratta di una tripla porta NOR a tre ingressi. La figura 12 riporta la tavola di verità delle porte NOR, mentre la figura 13 mostra il collegamento delle tre porte per realizzare la corretta interpretazio-

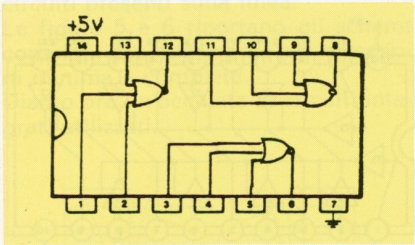


fig. 11 Il circuito integrato 74LS27

ne dei segnali provenienti dalla CPU. $\overline{IORQ}$ (richiesta di ingresso/uscita) e A5 (la linea del bus indirizzi che abbiamo scelto per indicare il nostro dispositivo) dovranno essere entrambi a livello basso, insieme a $\overline{WR}$ per attivare OKOUT,

a	b	c	out
0	0	0	1
0	0	1	0
0	1	0	0
0	1	1	0
1	0	1	0
1	1	0	0
1	1	1	0

fig. 12 Tavola di verità delle porte NOR

ed insieme a $\overline{RD}$ per attivare OKIN. Per quest'ultimo, infine, sarà necessaria un'inversione, operata dall'ultima porta i cui ingressi sono collegati insieme, a realizzare un inverter.

Il bus driver

Serve, nella versione completa, a pilotare il bus, dandogli maggiore potenza ed evitando che la CPU sia sottoposta ad un carico eccessivo.
Il circuito usato è un 74LS245, che vedete in figura 13.
Si tratta di un circuito sostanzialmente uguale a quello usato come controllore di ingresso, con la differenza che qui sono possibili due sensi di funzionamento. Il piedino 1, che comanda la se-

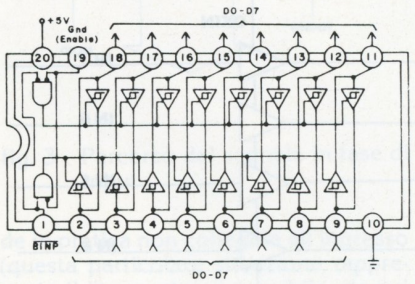


fig. 13 Il circuito integrato 74LS245

lezione della direzione di funzionamento, sarà normalmente a livello alto. In queste condizioni il circuito trasmette i segnali dalla CPU verso i controllori. Nel caso di una richiesta di input, OKIN porterà a livello basso il piedino 1, invertendo così il senso di trasmissione.
Le figure 14 e 15 riportano gli schemi dei collegamenti per le due versioni.
Dato il numero piuttosto elevato dei collegamenti, ci è sembrato utile riassumere quelli di ciascun integrato nelle figure 16,17,18 e 19. In questo modo potrete controllare più agevolmente l'esattezza del montaggio.
Per il montaggio vale quanto è stato detto la volta scorsa per il monitor-sonda logica. Ricordo a questo proposito che è indispensabile aver costruito sia questo che l'alimentatore, indispensabili alla messa a punto ed al funzionamento dell'interfaccia.
La piastra forata utilizzata per il montaggio misura cm 10x7, il doppio di quelle usate per gli altri due montaggi.
Per quanto riguarda il connettore la difficile reperibilità di un componente con le caratteristiche necessarie ci costringe ad un lavoro di "bricolage".
Dopo aver acquistato un connettore per schede che abbia 2.5 mm di passo dovremo segarne le due estremità, avendo cura di lasciare un numero minimo di 21 contatti. Il connettore che vedete nella foto ha tutti e 28 (27 più la tacca di centraggio) i contatti dello Spectrum; questo può essere utile se un giorno avessimo l'esigenza di creare dietro ad esso una possibilità di innesto per ulteriori periferiche.
Ora sceglieremo dove creare la tacca di centraggio, tenendo presente che alla sua sinistra devono esserci per lo meno due contatti. Con delle pinze sottili estrarremo i contatti ed al loro posto inseriremo un pezzo di plastica di dimensioni opportune.
Fatto questo, potremo saldare in posizione il connettore, esattamente come avremo fatto con gli zoccoli degli integrati. Fra il connettore e gli zoccoli, sulla piastra, dovranno esserci per lo meno due centimetri, per permettere l'inserimento completo.
In alto a sinistra dovrete montare un connettore per il collegamento con il monitor. Può essere di qualsiasi tipo, purché abbia 10 contatti (le 8 linee, il positivo ed il negativo). Aspettate invece, prima di collegare il suo corrispet-

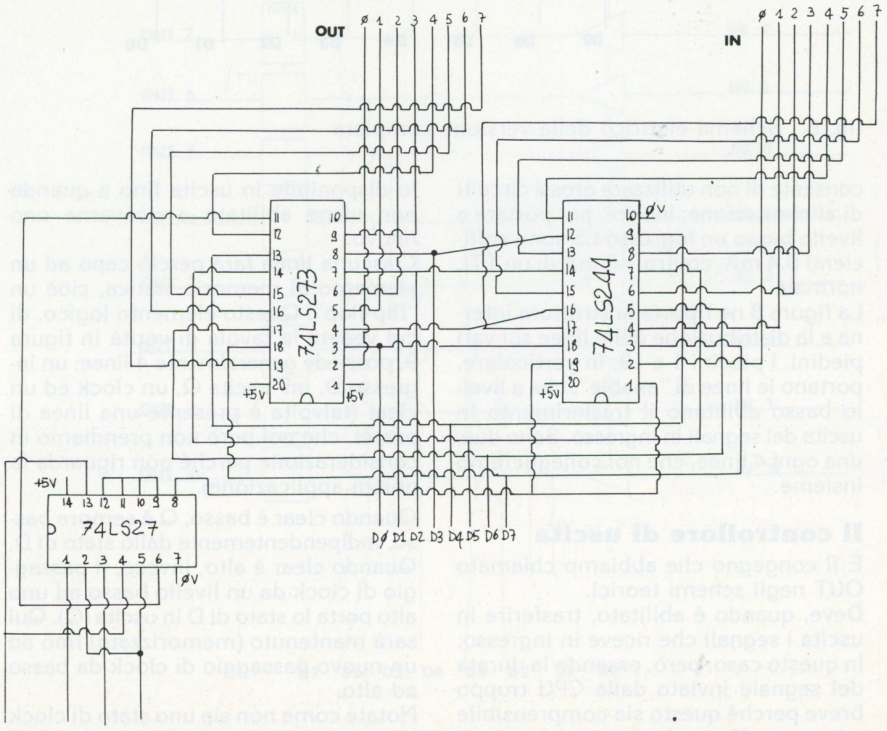


fig. 14 Schema dei collegamenti versione minima

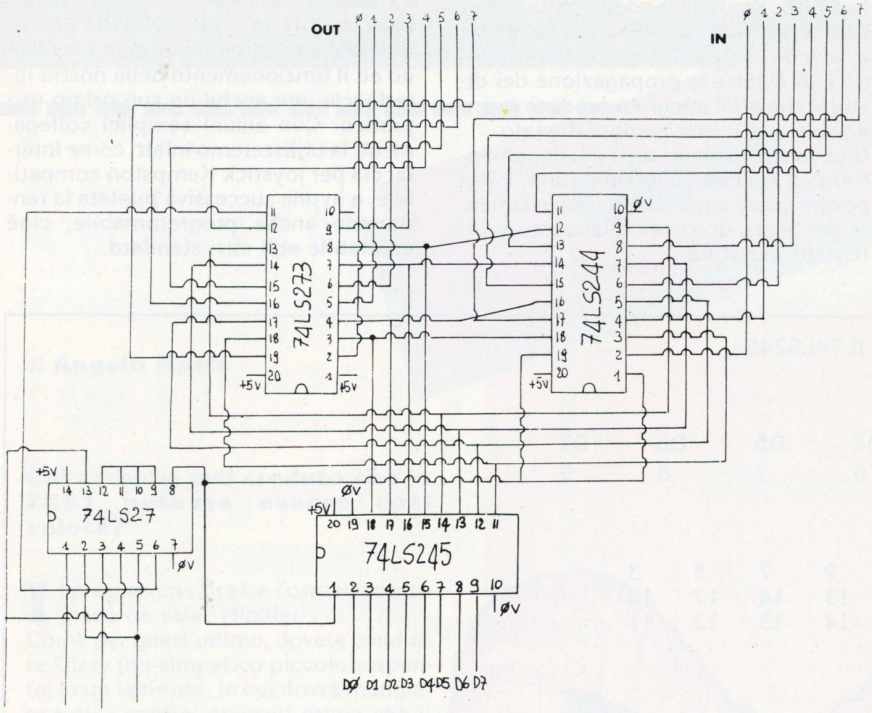


fig. 15 Schema dei collegamenti, versione completa

tivo sulla piastra del monitor, che il suo compito di sonda logica sia terminato. I componenti del circuito non sono stati elencati a parte, dato il loro numero

con la CPU:

RD	IORQ	WR	A5
1	2,5	3	4,13

con altri integrati:

74LS273-374		11
		11
74LS244	1	19
	8	8

alimentazione:

+	—
14	7

fig. 16 Tabella dei collegamenti per il 74LS27

davvero esiguo. Oltre ai circuiti integrati, 3 o 4 a seconda della versione, sono necessari i relativi zoccoli, a 20 piedini per il 74LS244, 74LS245, 74LS273 o 374 ed a 14 piedini per il 74LS27. Vi serviranno inoltre i due connettori

fig. 17 Tabella dei collegamenti per il 74LS244

con la CPU (versione minima) o con le uscite del 74LS245 (versione completa)

CPU	D0	D1	D2	D3	D4	D5	D6	D7
245	18	17	16	15	14	13	12	11
	18	16	14	12	9	7	5	3

entrate dall'esterno

IN0	IN1	IN2	IN3	IN4	IN5	IN6	IN7
2	4	6	8	11	13	15	17

alimentazione

+	—
20	10

con altri integrati

74LS27	8	8
	1	19

fig. 18 Tabella dei collegamenti per il 74LS273-374

con la CPU (versione minima) o con le uscite del 74LS245 (versione completa)

CPU	D0	D1	D2	D3	D4	D5	D6	D7
245	18	17	16	15	14	13	12	11
	3	4	7	8	13	14	17	18

uscite verso l'esterno

OUT0	OUT1	OUT2	OUT3	OUT4	OUT5	OUT6	OUT7
2	5	6	9	12	15	16	19

alimentazione

	+	—
273	1,20	10
374	20	1,10

con altri integrati

74LS27	11	
	11	

già descritti (se proprio volete, potrete saldare il monitor al circuito, risparmiando un connettore, ma non lo consiglio), un morsetto per l'alimentazione ed i fili per i collegamenti, delle dimensioni descritte nella scorsa puntata. È inoltre consigliabile procurarsi tanti condensatori da 0,1 microfarad quanti sono gli integrati utilizzati.
I condensatori, che non compaiono nello schema dei collegamenti, vanno collegati fra il positivo ed il negativo dell'alimentazione, il più vicino possibile a ciascun circuito integrato. Il loro scopo è di evitare la propagazione dei disturbi che ogni integrato produce quando una sua uscita cambia di stato.
Una volta terminati tutti i collegamenti fra gli zoccoli ed il connettore, e dopo che questi sono stati scrupolosamente verificati, occorre collaudare un integrato per volta.

Purtroppo, per ragioni di spazio, siamo costretti a terminare qui. Nella prossima puntata vedremo non solo il collaudo ed il funzionamento della nostra interfaccia, ma anche un suo primo uso pratico. Con alcuni semplici collegamenti la utilizzeremo infatti come interfaccia per joystick Kempston compatibile, e in una successiva puntata la renderemo anche programmabile, cioè adattabile agli altri standard.

fig. 19 Tabella dei collegamenti per il 74LS245

con la CPU

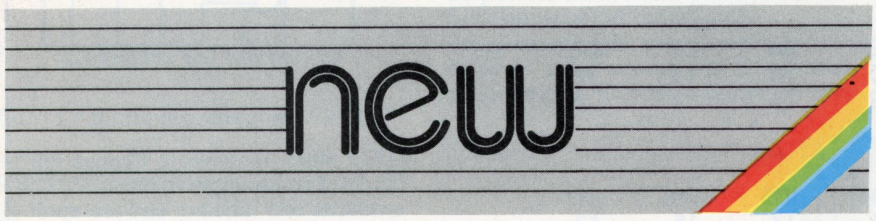
D0	D1	D2	D3	D4	D5	D6	D7
2	3	4	5	6	7	8	9

con altri integrati

74LS244		18	16	14	12	9	7	5	3
74LS273-374		3	4	7	8	13	14	17	18
		18	17	16	15	14	13	12	11
74LS27	8								
	1								

alimentazione

+	—
20	10,19



Vincenzo Garlaschelli

Direttore Generale della Sinclair Italia

Come già anticipato nello scorso numero, nelle prossime settimane verrà costituita ufficialmente la Sinclair Italia S.p.A., consociata della Sinclair Research.

Il direttore della nuova società, operativa a partire dall'anno prossimo e con sede a Cinisello Balsamo, è Vincenzo Garlaschelli, milanese trentanovenne, proveniente da incarichi dirigenziali nell'ambito di aziende quali Control Data Italia e Commodore Italiana.

La Sinclair Italia coprirà l'intero territorio nazionale sia per quanto riguarda la distribuzione sia - questo è confortante per molti sinclairisti - per l'assistenza tecnica. "La Sinclair Italia - ha dichiarato Garlaschelli - vuole inserirsi ai vertici del mercato italiano dei personal e home computer. Per raggiungere questo obiettivo intendiamo investire nella produzione di un cospicuo software applicativo in italiano, corredato da una documentazione chiara e precisa, ed attuare una politica di vendita che privilegi sia i distributori sia l'utente finale, garantendo "nel tempo" un servizio completo e la massima professionalità.

Gizzy

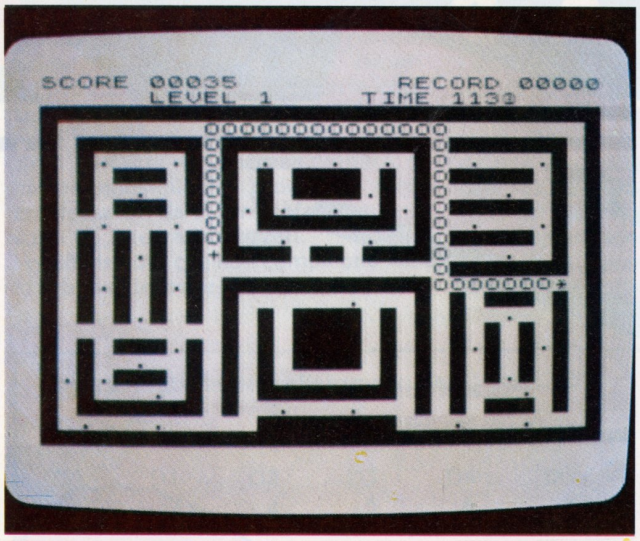
di **Angelo Motta**

Chi avrebbe mai creduto che lo ZX81 potesse essere così veloce?

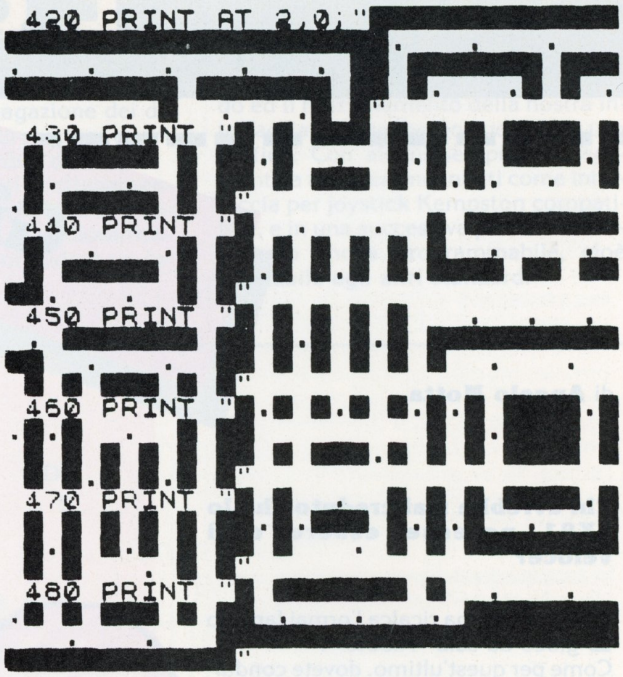
Il programma ricalca l'ormai famoso gioco da sala "Nibbler".
Come per quest'ultimo, dovete condurre Gizzy (un simpatico piccolo serpente) in un labirinto, in cui dovrà mangiare tutti i puntini presenti prima che il tempo a disposizione finisca. Ad ogni puntino mangiato il corpo di Gizzy si allungherà e dovrete fare attenzione che non rimanga imprigionato dal proprio corpo.
Il punteggio assegnato è di un punto per ogni puntino mangiato più, a labirinto completato, tanti punti quanto il tempo rimasto.
Sono previsti 9 livelli di gioco, uno più difficile dell'altro, ai quali corrisponde una velocità maggiore, un Gizzy sempre più lungo alla partenza e i puntini da mangiare che gradualmente aumentano.
Quando sarà stato completato anche il nono livello (molto difficile, considerata soprattutto la velocità del movimento) il gioco ripartirà dal primo, mantenendo inalterato il punteggio sino a quel momento raggiunto.
Per guidare Gizzy attraverso il labirinto dovrete utilizzare i tasti con le frecce (5-6-7-8).
Come per la maggior parte dei programmi in L/M, anche questo gioco richiede la costruzione in due fasi successive.
Nella prima dovrete creare la REM di almeno 606 caratteri (e qui ognuno si diverta come vuole) ed inserire il caricatore del L/M come da listato "1".
Successivamente, dopo aver inserito i codici del L/M, cancellare le righe del caricatore e copiare il resto del programma BASIC.
Si raccomanda, prima di provare il programma, di registrarlo, in modo da mettersi al riparo da eventuali errori nel codice macchina che potrebbero bloccare lo ZX81, il cui unico sblocco possibile sarebbe lo spegnimento della macchina, con conseguente perdita del lavoro fatto.
Buon divertimento.

Gizzy versione per ZX81

```
    ( RtM ?? ?▌?▌?   ( ?▌??????X8
000E£RND" "  ▒6▙RND: +▒5▜RND: :▒6▚RN
D: :▒6▒RND" ▗ ▚▒5▙RND5▒RND0 70 "   T
9N ( RETURN 0▗( RETURN 0*$4 PRIN
T TAN  NUOVO RECORDTEMPO SCADUTO
E▗RND76▗RNDE▙RND0F/ PLOT W?TAN E
▚RND00E▙RND▗▜ 00F( CLS 5▗RND0 70
5▙INKEY$0RNDE▚RND0)=70▌ E▒RND00F0
0F05F015▗RND0 70 E▙RND6▗RND0+CHR
$ 0?▙LEN "( UNPLOT ?VAL )5  GOSU
B ?00 FAST FOR "  RETURN ?E▗RND77
5▗RND▒??? LPRINT AT ( STOP )5  G
OSUB ?0*6▒RNDY▙LN ACS RNDX4 IF U
▙INKEY$CHR$ ▛M▙INKEY$:T( RETURN_
$4 CLS LN ▙"?LEN " AND ?PI??LN ▗
▛RND RETURN 54▒F// RETURN 64▌▗ G
OSUB ?/? RETURN 74▌▗ GOSUB PI/▚"
RETURN 8TAB ?PI7?▗RND▒▒ FOR 5 RE
TURN ? GOSUB ?▗RND▒ GOSUB ?▙RND?
????00 FOR 0*6▙RNDLN LET RNDU▙RN
DX RETURN  4G GOSUB ?▙RND?? RETU
RN  C▒LN LET RND+4 IF ( SAVE LN
LET RND$4 IF E▚RND0?5▙INKEY$0RND
?▛ TAN M▙RND40 RETURN  4G0* FAST
 E▙RND0 E ?6▙RND0+5 "?) ? GOSUB ?
▙RND GOSUB ▙E▙RND00 FOR F?F? LPR
INT 6▙RND/▒ RETURN ▛4JE▒RND9F/ P
LOT X?E▙RNDF6▙RND▗"  GOSUB PIAB
S ▙INKEY$E▗RND" £ ▒) SLOW RND FOR
 ▒ GOSUB ▙/( FAST FOR "  RETURN ?
```




```
E RND ??? LPRINT  GOSUB PIC  FO
R Y E RNDLN ACS RNDX4 IF E RND G
OSUB ? RND  GOSUB PIK3 GOSUB ?  R
ND" ( E RND :£)**RND FOR GOSUB
 E RND GOSUB ? RND GOSUB  Y2( R
ETURN $4 CLS TAN
   2 REM (C) A.MOTTA - MONZA
 100 LET L=USR 16540
 110 POKE 16418,0

 120 POKE 16389,120
 130 PRINT AT 5,8;"
 ";TAB 8;"
 ";TAB
 8;"
 "
 140 PRINT AT 11,3;"VUOI LE ISTR
UZIONI? (S/N)"
 150 GOTO 150+(10 AND INKEY$="S"
)+(250 AND INKEY$="N")
 160 CLS
 170 PRINT TAB 12;"

 180 PRINT ,,"AIUTATE IL PICCOLO
 SERPENTE A   MANGIARE TUTTI I P
UNTINI DEL   LABIRINTO PRIMA CH
E IL TEMPO A  DISPOSIZIONE FINIS
CA E SENZA"

 190 PRINT "CHE RIMANGA IMPRIGIO
NATO FRA IL PROPRIO CORPO."
 200 PRINT "UN PUNTO PER OGNI PU
NTINO MAN-  GIATO PIU' IL TEMPO
RIMASTO AG- GIUNTO AL PUNTEGGIO"
 210 PRINT "AD OGNI LIVELLO VERR
ANNO AUMEN- TATE LA VELOCITA' E
LA LUNGHEZ- ZA INIZIALE DI GIZZY
."

 220 PRINT "RAGGIUNTO IL NONO LI
VELLO  IL   GIOCO RIPRENDERA' DA
L PRIMO."
 230 PRINT "PER MUOVERE GIZZY US
A I TASTI   CON LE FRECCE (5-6-7
-8)."
 240 PRINT ,,"BUON DIVERTIMENTO"
 250 PRINT ,,,,"NEW LINE PER INI
ZIARE"
 390 IF INKEY$<>CHR$ 118 THEN GO
TO 390
 400 CLS
 410 PRINT "SCORE 00000
RECORD 00000      LEVEL 0     TI
ME 0000"
```



```
 420 PRINT AT 2,0;"
 430 PRINT "
 440 PRINT "
 450 PRINT "
 460 PRINT "
 470 PRINT "
 480 PRINT "
 490 POKE 16534,77
 500 IF L<31 THEN GOTO 700
 510 PRINT AT 3,3;CHR$ 27;AT 3,1
1;CHR$ 27;AT 3,27;CHR$ 27;AT 5,5
;CHR$ 27;AT 5,22;CHR$ 27;AT 10,1
1;CHR$ 27;AT 10,20;CHR$ 27;AT 11
,3;CHR$ 27
 520 PRINT AT 11,24;CHR$ 27;AT 1
4,15;CHR$ 27;AT 15,7;CHR$ 27;AT
15,28;CHR$ 27;AT 20,23;CHR$ 27;A
T 22,4;CHR$ 27;AT 21,15;CHR$ 27
 530 POKE 16534,92
 540 IF L<33 THEN GOTO 700
 550 PRINT AT 5,1;CHR$ 27;AT 17,
1;CHR$ 27;AT 9,5;CHR$ 27;AT 9,9;
CHR$ 27;AT 17,9;CHR$ 27;AT 6,13;
CHR$ 27;AT 19,13;CHR$ 27;AT 6,20
;CHR$ 27;AT 15,24;CHR$ 27
 560 PRINT AT 22,21;CHR$ 27;AT 7
,7;CHR$ 27;AT 20,4;CHR$ 27;AT 22
,30;CHR$ 27;AT 17,22;CHR$ 27;AT
19,20;CHR$ 27;AT 7,28;CHR$ 27;AT
7,3;CHR$ 27;AT 3,19;CHR$ 27
 570 POKE 16534,110
 580 IF L<35 THEN GOTO 700
 590 PRINT AT 22,8;CHR$ 27;AT 19
,7;CHR$ 27;AT 16,3;CHR$ 27;AT 13
,1;CHR$ 27;AT 9,1;CHR$ 27;AT 21,
1;CHR$ 27;AT 3,7;CHR$ 27;AT 3,15
;CHR$ 27;AT 3,23;CHR$ 27
 600 PRINT AT 19,17;CHR$ 27;AT 1
0,15;CHR$ 27;AT 17,11;CHR$ 27;AT
 12,10;CHR$ 27;AT 11,7;CHR$ 27;A
T 9,26;CHR$ 27;AT 17,26;CHR$ 27;
AT 22,25;CHR$ 27;AT 11,17;CHR$ 2
7
 610 POKE 16534,128
 700 LET L=USR (16648+(28 AND L<
>0))
 720 IF L<>0 THEN GOTO 420
 730 PRINT AT 5,11;"          ";
TAB 11;"GAME  OVER";TAB 11;"
          ";AT 12,5;"NEW LINE PER AL
TRA GARA"
 740 IF INKEY$<>CHR$ 118 THEN GO
TO 740
 750 GOTO 420
 800 STOP
 998 SAVE "GIZZ"
 999 RUN
```

Gizzy

```
  L REM CHE DEVE ESSERE SEGUITA
 DA ALMENO 606 CARATTERI (DALLA
LOCAZIONE 16514 ALLA 17119)
 100 FOR I=16540 TO 17119
 110 INPUT A
 120 IF PEEK 16418=2 THEN SCROLL

 130 PRINT I;" = ";A
 140 POKE I,A
 150 NEXT I
 160 REM DARE IL RUN ED INSERIRE
I CODICI DEL L/M DEL LISTATO 2.
 170 REM AD INSERIMENTO FINITO
```

IL PROGRAMMA SI FERMERA".

```
 180 REM CANCELLARE LE RIGHE DA
100 A 180 ED INSERIRE IL RESTO.
DEL PROGRAMMA COME DA LISTATO 3.
```

```
16540 = 42
16541 = 12
16542 = 64
16543 = 1
16544 = 11
16545 = 0
16546 = 9
16547 = 34
16548 = 130
16549 = 64
16550 = 14
16551 = 21
16552 = 9
16553 = 34
16554 = 132
16555 = 64
16556 = 14
16557 = 14
16558 = 9
16559 = 34
16560 = 134
16561 = 64
16562 = 14
16563 = 14
16564 = 9
16565 = 34
16566 = 136
16567 = 64
16568 = 1
16569 = 185
16570 = 2
16571 = 9
16572 = 34
16573 = 144
16574 = 64
16575 = 33
16576 = 138
16577 = 64
16578 = 54
16579 = 0
16580 = 35
16581 = 54
16582 = 0
16583 = 1
16584 = 0
16585 = 0
16586 = 201
16587 = 16
16588 = 254
16589 = 54
16590 = 151
16591 = 16
16592 = 254
16593 = 54
16594 = 23
16595 = 13
16596 = 32
16597 = 245
16598 = 201
16599 = 0
16600 = 51
16601 = 58
16602 = 52
16603 = 59
16604 = 52
16605 = 0
16606 = 55
16607 = 42
16608 = 40
16609 = 52
16610 = 55
16611 = 41
16612 = 57
16613 = 42
16614 = 50
16615 = 53
16616 = 52
16617 = 0
16618 = 56
16619 = 40
16620 = 38
16621 = 41
16622 = 58
16623 = 57
16624 = 52
16625 = 42
16626 = 140
16627 = 64
16628 = 35
16629 = 34
16630 = 140
16631 = 64
16632 = 42
16633 = 130
16634 = 64
16635 = 126
16636 = 254
16637 = 37
16638 = 32
16639 = 5
16640 = 54
16641 = 28
16642 = 43
16643 = 24
16644 = 246
16645 = 60
16646 = 119
16647 = 201
16648 = 42
16649 = 134
16650 = 64
16651 = 54
16652 = 28
16653 = 42
16654 = 130
16655 = 64
16656 = 6
16657 = 5
16658 = 54
16659 = 28
16660 = 43
16661 = 16
16662 = 251
16663 = 33
16664 = 140
16665 = 64
16666 = 54
16667 = 0
16668 = 35
16669 = 54
16670 = 0
16671 = 33
16672 = 145
16673 = 65
16674 = 54
16675 = 64
16676 = 42
16677 = 134
16678 = 64
16679 = 126
16680 = 60
16681 = 119
16682 = 33
16683 = 142
16684 = 64
16685 = 54
16686 = 220
16687 = 35
16688 = 54
16689 = 5
16690 = 42
16691 = 136
16692 = 64
16693 = 54
16694 = 28
16695 = 43
16696 = 54
16697 = 28
16698 = 43
16699 = 54
16700 = 33
16701 = 43
16702 = 54
16703 = 29
16704 = 33
16705 = 151
16706 = 64
16707 = 54
16708 = 0
16709 = 35
16710 = 54
16711 = 0
16712 = 42
16713 = 144
16714 = 64
16715 = 34
16716 = 148
16717 = 64
16718 = 54
16719 = 21
16720 = 214
16721 = 28
16722 = 71
16723 = 175
16724 = 198
16725 = 2
16726 = 16
16727 = 252
16728 = 71
16729 = 197
16730 = 17
16731 = 33
16732 = 0
16733 = 237
16734 = 82
16735 = 54
16736 = 52
16737 = 229
16738 = 235
16739 = 1
16740 = 254
16741 = 119
16742 = 42
16743 = 151
16744 = 64
16745 = 35
16746 = 35
16747 = 34
16748 = 151
16749 = 64
16750 = 9
16751 = 115
16752 = 35
16753 = 114
16754 = 225
16755 = 193
16756 = 16
16757 = 227
16758 = 17
16759 = 33
16760 = 0
16761 = 237
16762 = 82
16763 = 54
16764 = 23
16765 = 34
16766 = 146
16767 = 64
16768 = 62
16769 = 5
16770 = 205
16771 = 203
16772 = 64
16773 = 61
16774 = 32
16775 = 250
16776 = 58
16777 = 145
16778 = 65
16779 = 214
16780 = 7
16781 = 50
16782 = 145
16783 = 65
16784 = 14
16785 = 57
16786 = 16
16787 = 254
16788 = 13
16789 = 32
16790 = 251
16791 = 205
16792 = 187
16793 = 2
16794 = 124
16795 = 198
16796 = 2
16797 = 218
16798 = 96
16799 = 66
16800 = 68
16801 = 77
```

```
16802 = 205
16803 = 189
16804 = 7
16805 = 126
16806 = 1
16807 = 33
16808 = 0
16809 = 42
16810 = 146
16811 = 64
16812 = 254
16813 = 33
16814 = 32
16815 = 3
16816 = 43
16817 = 24
16818 = 24
16819 = 254
16820 = 34
16821 = 32
16822 = 5
16823 = 167
16824 = 237
16825 = 74
16826 = 24
16827 = 15
16828 = 254
16829 = 35
16830 = 32
16831 = 5
16832 = 167
16833 = 237
16834 = 66
16835 = 24
16836 = 6
16837 = 254
16838 = 36
16839 = 194
16840 = 96
16841 = 66
16842 = 35
16843 = 126
16844 = 254
16845 = 27
16846 = 32
16847 = 92
16848 = 237
16849 = 75
16850 = 151
16851 = 64
16852 = 3
16853 = 3
16854 = 235
16855 = 33
16856 = 254
16857 = 119
16858 = 237
16859 = 67
16860 = 151
16861 = 64
16862 = 9
16863 = 237
16864 = 75
16865 = 146
16866 = 64
16867 = 113
16868 = 35
16869 = 112
16870 = 96
16871 = 105
16872 = 54
16873 = 52
16874 = 235
16875 = 54
16876 = 23
16877 = 34
16878 = 146
16879 = 64
16880 = 205
16881 = 241
16882 = 64
16883 = 58
16884 = 150
16885 = 64
16886 = 61
16887 = 254
16888 = 0
16889 = 32
16890 = 44
16891 = 237
16892 = 75
16893 = 142
16894 = 64
16895 = 87
16896 = 120
16897 = 254
16898 = 0
16899 = 40
16900 = 8
16901 = 205
16902 = 241
16903 = 64
16904 = 21
16905 = 32
16906 = 250
16907 = 16
16908 = 248
16909 = 205
16910 = 241
16911 = 64
16912 = 13
16913 = 32
16914 = 250
16915 = 42
16916 = 134
16917 = 64
16918 = 126
16919 = 254
16920 = 37
16921 = 32
16922 = 8
16923 = 62
16924 = 28
16925 = 119
16926 = 33
16927 = 145
16928 = 65
16929 = 54
16930 = 64
16931 = 79
16932 = 6
16933 = 0
16934 = 201
16935 = 50
16936 = 150
16937 = 64
16938 = 32
16939 = 52
16940 = 254
16941 = 0
16942 = 32
16943 = 44
16944 = 54
16945 = 23
16946 = 229
16947 = 42
16948 = 148
16949 = 64
16950 = 54
16951 = 0
16952 = 42
16953 = 0
16954 = 120
16955 = 34
16956 = 148
16957 = 64
16958 = 64
16959 = 21
16960 = 33
16961 = 2
16962 = 120
16963 = 17
16964 = 0
16965 = 120
16966 = 237
16967 = 75
16968 = 151
16969 = 64
16970 = 237
16971 = 176
16972 = 42
16973 = 146
16974 = 64
16975 = 54
16976 = 52
16977 = 235
16978 = 43
16979 = 114
16980 = 43
16981 = 115
16982 = 225
16983 = 34
16984 = 146
16985 = 64
16986 = 24
16987 = 3
16988 = 254
16989 = 128
16990 = 32
16991 = 47
16992 = 42
16993 = 136
16994 = 64
16995 = 126
16996 = 254
16997 = 28
16998 = 32
16999 = 5
17000 = 54
17001 = 37
17002 = 43
17003 = 24
17004 = 246
17005 = 61
17006 = 119
17007 = 42
17008 = 142
17009 = 64
17010 = 43
17011 = 34
17012 = 142
17013 = 64
17014 = 167
17015 = 1
17016 = 1
17017 = 0
17018 = 237
17019 = 66
17020 = 210
17021 = 144
17022 = 65
17023 = 42
17024 = 144
17025 = 64
17026 = 1
17027 = 12
17028 = 0
17029 = 9
17030 = 17
17031 = 228
17032 = 64
17033 = 235
17034 = 3
17035 = 237
17036 = 176
17037 = 24
17038 = 19
17039 = 229
17040 = 235
17041 = 1
17042 = 254
17043 = 119
17044 = 42
17045 = 151
17046 = 64
17047 = 9
17048 = 78
17049 = 35
17050 = 70
17051 = 225
17052 = 167
17053 = 237
17054 = 66
17055 = 40
17056 = 191
17057 = 235
17058 = 62
17059 = 5
17060 = 42
17061 = 146
17062 = 64
17063 = 205
17064 = 203
17065 = 64
17066 = 61
17067 = 32
17068 = 250
17069 = 42
17070 = 138
17071 = 64
17072 = 237
17073 = 75
17074 = 140
17075 = 64
17076 = 167
17077 = 237
17078 = 66
17079 = 48
17080 = 31
17081 = 237
17082 = 67
17083 = 138
17084 = 64
17085 = 1
17086 = 16
17087 = 0
17088 = 42
17089 = 136
17090 = 64
17091 = 9
17092 = 14
17093 = 12
17094 = 17
17095 = 216
17096 = 64
17097 = 235
17098 = 237
17099 = 176
17100 = 1
17101 = 5
17102 = 0
17103 = 42
17104 = 130
17105 = 64
17106 = 237
17107 = 91
17108 = 132
17109 = 64
17110 = 237
17111 = 184
17112 = 62
17113 = 30
17114 = 16
17115 = 254
17116 = 13
17117 = 32
17118 = 251
17119 = 201
```

Sul prossimo numero il listato ASSEMBLER di GIZZY Non perdetelo!

Pallonzolo

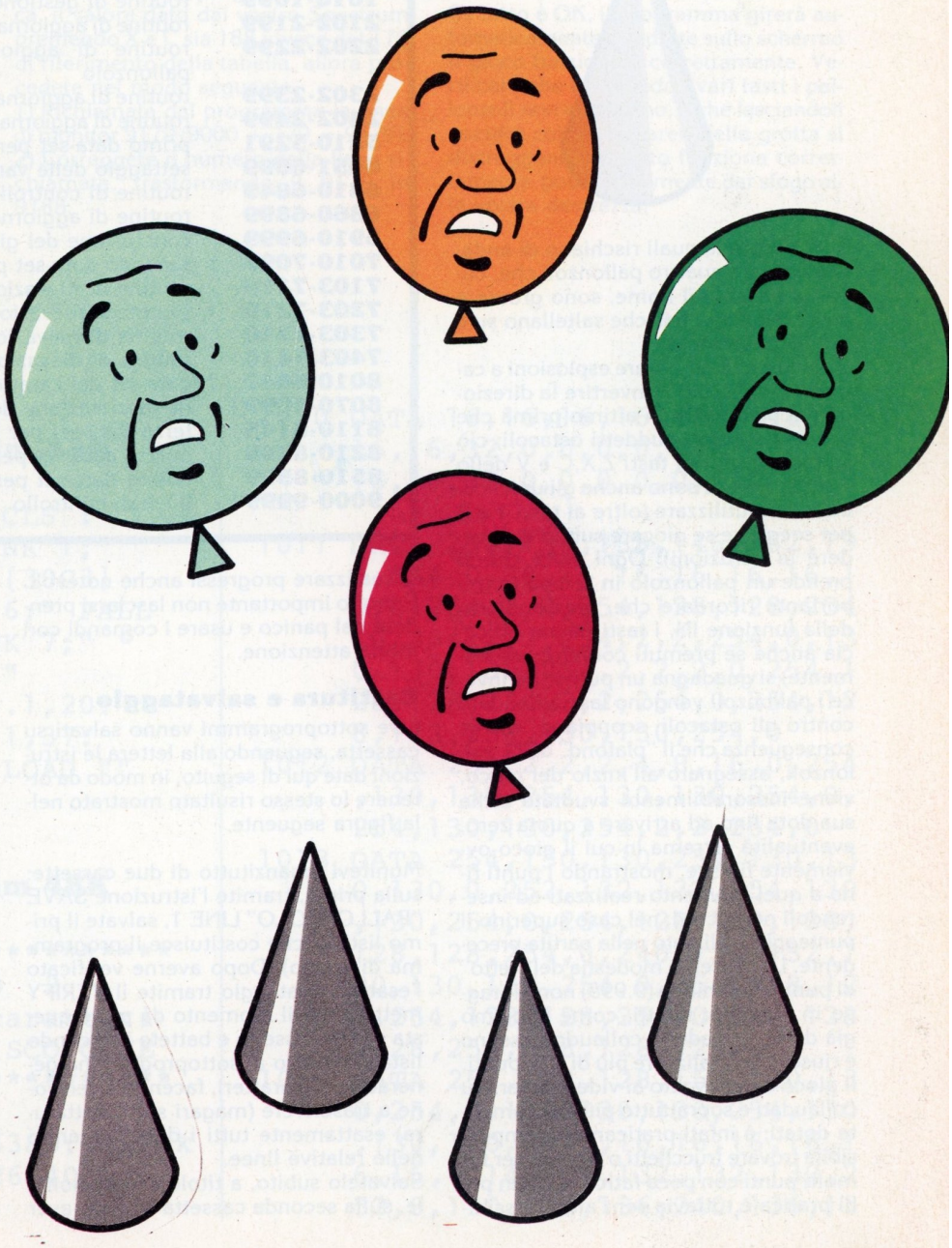
di **Daniele Malavasi**

Una produzione per SUPERSINC DiscoVogue HARDWARE AND SOFTWARE DIFFUSION

Questo programma è stato creato appositamente per quella nutrita schiera di spectromani che amano dilettarsi con i jogging, cioè con quei particolari videogame che mettono a dura prova non solo riflessi e abilità, ma anche e soprattutto calma e tenacia, doti con le quali un giocatore riesce a far fronte a situazioni caotiche o che da facili diventano improvvisamente incontrollabili. Si tratta di un game che mette alle corde chi di solito impara subito i trucchetti necessari a far molti punti con poca fatica; o per lo meno, se ciò può essere possibile anche con Pallonzolo, può costare molti sacrifici di applicazione e di pratica. Da ricordare che nessuno dei collaudatori designati, dopo ore di costante impegno, è riuscito a passare quota 118 punti (sui 9.999 disponibili).
Il programma contiene accorgimenti e funzioni di BASIC esteso piuttosto particolari, per cui potrà essere interessante anche come supporto didattico integrativo: scorrendo attentamente le linee del listato 3 troverete le funzioni IN, STR$, CODE INKEY$ ed altre che non trovano diffusa applicazione in programmi di questo tipo.
Da ricordare che il programma in oggetto, occupando circa 13 Kbyte di memoria, non potrà girare sugli Spectrum "base" di 16 Kbyte.

La strategia di gioco

Il programma, attraverso istruzioni illustrate molto chiaramente e con una grafica di video molto efficace, spiega meglio di qualsiasi discorso sia le caratteristiche di base che gli obiettivi da raggiungere. Le istruzioni di gioco possono essere viste, volendo, sia all'inizio della prima partita che all'inizio di ogni competizione successiva.
Pallonzolo è ambientato in una grotta riccamente dotata di stalattiti e stalag-



Descrizione del programma

Linee	Commento
Listato 1: SOTTOPROGRAMMA DI LANCIO	
10	inizio del sottoprogramma; preparazione del video
11	dati preliminari per il caricamento dei sottoprogrammi successivi
Listato 2: GENERATORE DI CARATTERI	
990-1000	inizio del sottoprogramma; dati preliminari
1001	spostamento del set dei caratteri grafici all'indirizzo 42000 della RAM
1010-1016	data-set per la generazione dei nuovi caratteri grafici
1099-1104	inserimento dei nuovi caratteri grafici all'indirizzo 42000 della RAM
1190	elementi di controllo e verifica
Listato 3: MASTER	
10-20	predisposizione video; dati preliminari
510-580	preparazione grafica del video
610-680	istruzioni di gioco
1010-1099	routine di gestione principale
2102-2199	routine di aggiornamento e stampa del primo pallonzolo
2202-2299	routine di aggiornamento e stampa del secondo pallonzolo
2302-2399	routine di aggiornamento e stampa del terzo pallonzolo
2402-2499	routine di aggiornamento e stampa del quarto pallonzolo
5210-5291	primo data-set per la gestione grafica
6051-6099	settaggio delle variabili di gioco e del video
6810-6849	routine di controllo dei pallanzoli
6860-6899	routine di aggiornamento del punteggio
6910-6999	conclusione del gioco; registro del record
7010-7099	secondo data-set per la gestione grafica
7103-7110	routine di rilevazione di errori per il primo pallonzolo
7203-7210	routine di rilevazione di errori per il secondo pallonzolo
7303-7310	routine di rilevazione di errori per il terzo pallonzolo
7403-7410	routine di rilevazione di errrori per il quarto pallonzolo
8010-8049	data-set dei caratteri grafici
8070-8099	memorizzazione delle stringhe di controllo
8110-8145	terzo data-set per la gestione grafica
8210-8299	quarto data-set per la gestione grafica
8510-8599	quinto data-set per la gestione grafica
9000-9999	linee di controllo

miti, contro le quali rischiano di andare a cozzare quattro pallonzoli che, come fa pensare il nome, sono grosse e pesanti palle di gas che saltellano su e giú continuamente.
L'unico modo di evitare esplosioni a catena è quello di far invertire la direzione dei pallonzoli un attimo prima che arrivino contro i suddetti ostacoli; ciò va fatto tramite i tasti Z,X,C e V dello Spectrum, che sono anche gli unici comandi da utilizzare (oltre ai tasti 1 e 2 per scegliere se giocare subito o se vedere le istruzioni). Ogni volta che si prende un pallonzolo in tempo (è importante ricordare che, grazie all'uso della funzione IN, i tasti hanno efficacia anche se premuti contemporaneamente) si guadagna un punto. Se invece i pallonzoli vengono lasciati andare contro gli ostacoli, scoppiano, con la conseguenza che il "plafond" di 21 pallonzoli, assegnato all'inizio del gioco, viene inesorabilmente svuotato della sua dote fino ad arrivare a quota zero, eventualità estrema in cui il gioco ovviamente finisce, mostrando i punti fino a quel momento realizzati ed inserendoli nel record, nel caso superino il punteggio realizzato nella partita precedente. L'apparente modestia del "tetto" di punti disponibile (9.999) non vi tragga in inganno; infatti, come abbiamo già detto, in sede di collaudo nessuno è riuscito a totalizzare più di 118 punti.
Il gioco è indirizzato ai videoplayer più collaudati e soprattutto più tatticamente dotati: è infati praticamente impossibile trovare trucchetti o sistemi per far molti punti con poca fatica. Con un po' di pratica è tuttavia senz'altro possibile realizzare progressi anche notevoli. È molto importante non lasciarsi prendere dal panico e usare i comandi con molta attenzione.

Battitura e salvataggio

I tre sottoprogrammi vanno salvati su cassetta, seguendo alla lettera le istruzioni date qui di seguito, in modo da ottenere lo stesso risultato mostrato nella figura seguente.

Munitevi innanzitutto di due cassette: sulla prima, tramite l'istruzione SAVE "PALLONZOLO" LINE 1, salvate il primo listato (che costituisce il programma di lancio). Dopo averne verificato l'esatto salvataggio tramite il VERIFY mettete per il momento da parte questa prima cassetta e battete il secondo listato relativo al sottoprogramma generatore di caratteri, facendo attenzione a trascrivere (magari sotto dettatura) esattamente tutti i data contenuti nelle relative linee.
Salvatelo subito, a titolo precauzionale, sulla seconda cassetta e, dopo aver controllato la registrazione col solito VERIFY, fate girare mediante RUN il programma che ancora avete sullo Spectrum: lo schermo deve allora rimanere completamente bianco per circa 16 secondi, dopodiché il messaggio 0 OK, 1190:3 segnalerà, in assenza di errori, l'esatto inserimento dei caratteri all'indirizzo 42000 della RAM. Attenzione: non è detto che comparendo detto messaggio vada sicuramente tutto bene, in quanto potrebbero essere inseriti dei data diversi da quelli di riferimento del listato 2. È indispensabile allora, più che mai, trascrivere alla perfezione tutti i numeri di data, pena la futura comparsa di caratteri indefiniti od altre stranezze grafiche.
Ammesso comunque che tutto vada per il verso giusto, potete salvare questo secondo listato sulla prima cassetta, lasciando circa 3-4 secondi di nastro vuoto dopo il sottoprogramma di lancio precedentemente salvato. L'istruzione da usare è SAVE "CARATTERI" CODE 42000, 1024. Infatti è necessario caricare i dati sotto forma di sequenza by-

te, allo scopo di accorciare il tempo di carica del gioco. In questo modo il secondo sottoprogramma, trasformato da BASIC in sequenza byte, viene ridotto da 3597 a 1024 byte e il relativo tempo di carica da 25 a 9 secondi.
A questo punto è possibile battere il terzo listato (e cioè il master); prima però si rende necessario vedere se la funzione IN del vostro Spectrum dà valori originali identici a quelli della tabella di riferimento. Infatti, non tutti gli Spectrum derivano gli stessi valori dalla funzione IN, ma per fortuna le caratteristiche strutturali del programma consentono di adattarlo ad ogni tipo di Spectrum; sono infatti previste delle apposite linee di monitor (linee di gosub dalla 9000 alla 9999), che possono essere adattate al vostro caso, seguendo scrupolosamente il seguente procedimento.
Impostate sullo Spectrum il seguente programma di verifica:

```
10 LET n= IN 65278
20 PRINT AT 0,0;" "; AT 0,0;n
30 BEEP .01, 50: GO TO 10
```

Digitando adesso tutte le sedici possibili combinazioni (mediante i tasti Z, X, C e V) verificate che i valori del vostro Spectrum siano uguali a quelli di riferimento. Per ogni valore che eventualmente risultasse diverso da quelli di riferimento in tabella effettuate sul master le seguenti operazioni.
1) Richiamate dal programma la linea di monitor relativa al valore di riferimento della tabella (il numero di linea si ottiene sommando 9.000 al numero di riferimento considerato).
2) Correggete il numero della linea di monitor richiamata, inserendo il nuovo numero di linea (che sarà ottenuto sommando a 9.000 il valore fornito dal vostro Spectrum con il programmino di verifica visto prima).
3) Se la nuova linea di monitor appena inserita non ha quella immediatamente precedente, allora inseritene una, scrivendo in essa sempre e solo RETURN.
4) Eliminate la vecchia linea di monitor.
L'esempio seguente toglierà ogni dubbio anche a coloro che non avessero ben capito tutto il procedimento: supponendo che, col programma di verifica, il valore dato dal vostro Spectrum premendo X e C sia 188 invece del 179 di riferimento della tabella, allora procedete nel modo seguente.
1) Richiamate dal programma la linea di monitor 9179 (9000 + 179 = 9179).
2) Correggete il numero della linea richiamata, trasformando il 9179 in 9188.
3) Se non esiste la linea 9187 (cioè quella immediatamente precedente alla 9188), allora inseritela scrivendo: 9187 RETURN.
4) Eliminate la linea 9179 (vecchia linea di monitor).

Tutto qui: effettuando attentamente e con un minimo di attenzione tutti i passaggi appena esposti, non dovreste trovare alcuna difficoltà. A questo punto si può procedere a salvare il master, originale o riadattato che sia, sulla cassetta, subito dopo il generatore di caratteri (senza lasciare spazi di nastro vuoto).

Adesso si può riavvolgere la cassetta e caricare tutto il programma con LOAD "".

Se tutto è OK, il programma girerà automaticamente e vedrete sullo schermo il gioco funzionare correttamente. Verificate che azionando i vari tasti i pallonzoli non esplodano, e che lasciandoli cozzare contro le pareti della grotta si distruggano. Se tutto funziona correttamente, potete finalmente dar sfogo alla vostra destrezza.

Pallonz.1

pallonzolo
versione per ZX Spectrum 48K
Listato 1

```
10 PAPER 0: BORDER 0: CLS :
   PRINT #1;AT 0,0; INK 1;"
   {G5}{30G3}{SG5}{G1}{30G3}
   {G2}";AT 0,10; INK 6;" PALL
   ONZOLO ";AT 1,5; INK 7;" ©
   1984 by DISCOVOGUE "
11 FOR n=1 TO 3: BEEP .1,20:
   BEEP .1,23: BEEP .1,37:
   NEXT n: PAUSE 30: LOAD ""
   CODE : LOAD ""
```

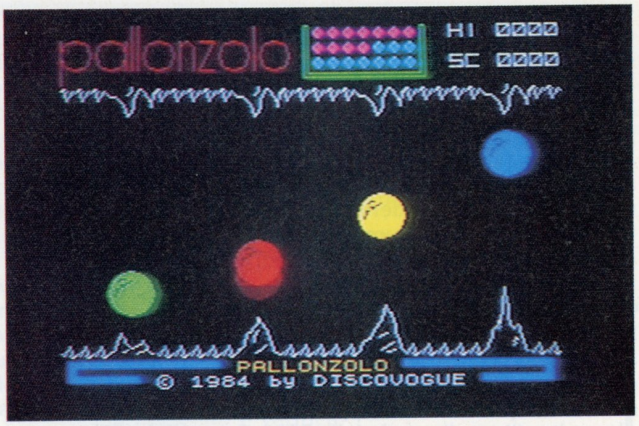
pallonzolo
versione per ZX Spectrum 48K
Listato 2

```
 990 REM **********************
 991 REM        LISTATO  2
 992 REM  generatore caratteri
 993 REM  © 1984 by DISCOVOGUE
 994 REM **********************
1000 POKE 23562,1
1001 FOR n=15360 TO 16383: LET k
     =PEEK n: POKE n+26640,k:
     NEXT n
1010 DATA 16,16,16,16,48,0,48,0,
     16,16,32,0,0,0,0,0,0,0,0,0,
     16,16,32,0,0,0,0,0,0,0,48,0
1011 DATA 254,134,138,146,162,19
     4,254,0,48,16,16,16,16,16,1
     6,0,254,2,2,254,128,128,254
     ,0,254,2,2,254,2,2,254,0,13
     0,130,130,254,2,2,2,0,254,1
     28,128,254,2,2,254,0,254,12
     8,128,254,130,130,254,0
1012 DATA 254,2,2,2,4,8,16,0,254
     ,130,130,254,130,130,254,0,
     254,130,130,254,2,2,254,0
1013 DATA 254,130,130,254,130,13
     0,130,0,252,132,132,254,130
     ,130,254,0,254,128,128,128,
     128,128,254,0,240,136,132,1
     30,130,130,254,0,254,128,12
     8,254,128,128,254,0,254,128
     ,128,254,128,128,128,0
1014 DATA 254,128,128,158,130,13
     0,254,0,130,130,130,254,130
     ,130,130,0,16,16,16,16,16,1
     6,16,0,14,4,4,132,132,132,2
     52,0,130,132,136,240,136,13
```

```
     2,130,0,128,128,128,128,128
     ,128,254,0,254,146,146,146,
     130,130,130,0,242,146,146,1
     46,146,146,158,0
1015 DATA 254,130,130,130,130,13
     0,254,0,254,130,130,254,128
     ,128,128,0,254,130,130,146,
     138,134,254,0,254,130,130,2
     54,144,144,158,0,254,128,12
     8,254,2,2,254,0,254,16,16,1
     6,16,16,16,0
1016 DATA 130,130,130,130,130,13
     0,254,0,130,130,68,68,40,40
     ,16,0,130,130,130,146,146,1
     46,254,0,130,68,40,16,40,68
     ,130,0,130,130,130,254,16,1
     6,16,0,254,2,4,56,64,128,25
     4,0
1099 RESTORE 1010: FOR n=42264
      TO 42271: READ k: POKE n,k
     : NEXT n
1100 FOR n=42312 TO 42319:
      READ k: POKE n,k: NEXT n
1101 FOR n=42352 TO 42359:
      READ k: POKE n,k: NEXT n
1102 FOR n=42368 TO 42375:
      READ k: POKE n,k: NEXT n
1103 FOR n=42384 TO 42463:
      READ k: POKE n,k: NEXT n
1104 FOR n=42520 TO 42727:
      READ k: POKE n,k: NEXT n
1190 POKE 23606,16: POKE 23607,1
     64: BEEP 1,30
```

pallonzolo
versione per ZX Spectrum 48K
Listato 3

```
 10 REM **********************
 11 REM        LISTATO  1
 12 REM        PALLONZOLO
 13 REM   © 1984 by DISCOVOGUE
 14 REM **********************
 15 POKE 23606,0: POKE 23607,60

 20 PAPER 0: BORDER 0: INK 7:
     CLS : LET hi=0: GO SUB 805
     0: GO SUB 8500: GO SUB 8000
     : POKE 23562,1
510 GO SUB 600: GO SUB 7000
580 GO TO 990
610 LET pp=13: LET i=3: PRINT
     AT 11,0; INK 5;"1 ........
```




```
     .. PER GIOCARE SUBITO"'"2
     ..... PER VEDERE LE ISTRUZI
     ONI"
611 LET k=CODE INKEY$: IF k=49
      THEN  GO TO 680
612 IF k=50 THEN  GO TO 620
613 LET pp=24-pp: PRINT AT pp,0
     ; INK 7; OVER 1; BRIGHT 1;a
     $;AT 24-pp,0; INK 3;
      BRIGHT 0;a$: BEEP .01,50:
      BEEP .1,60: GO TO 611
620 FOR n=11 TO 21: PRINT AT n,
     0;a$: BEEP .001,n*3: NEXT n

621 DATA 1,26,3,27,54,2,55,84,1
     ,85,110,3,111,132,5,133,162
     ,1,163,192,1,193,224,0,1,32
     ,0,33,60,2,61,92,0,93,124,0
     ,4,2,6,1
622 RESTORE 621: INK 5: FOR n=5
      TO 12: READ ii: READ ff:
      READ cl: PRINT AT n,cl;y$(
     ii TO ff): BEEP .01,n*3:
      NEXT n
623 FOR n=13 TO 16: READ i:
      READ f: READ cl: PRINT
     AT n,cl;t$(i TO f): BEEP .0
     1,n*3: NEXT n: INK 7
624 LET pp=4: PRINT AT 19,1;
      INK 7;"Z";AT 19,9;"X";AT 1
     9,17;"C";AT 19,25;"V":
      FOR n=3 TO 27 STEP 8:
      READ i: PRINT AT 18,n;
      INK i;d$;AT 19,n;e$;AT 20,
     n;f$: BEEP .1,n: NEXT n
670 IF INKEY$<>"" THEN  FOR n
     =4 TO 21: PRINT AT n,0;a$:
      BEEP .001,n*3: NEXT n:
      GO SUB 6000: RETURN
671 LET pp=10-pp: PRINT AT 16,0
     ; INK pp; BRIGHT 1; OVER 1;
     a$: BEEP .1,-pp: GO TO 670
```

```
 680 FOR n=11 TO 21: PRINT AT n,
     0;a$: BEEP .001,n*3:
      PAUSE 1: NEXT n: GO SUB 60
     00: RETURN
1010 LET k=IN 65278: LET z1=0:
      LET z2=0: LET z3=0: LET z4
     =0: GO SUB 9000+k
1030 BEEP .001,58: GO SUB 2100:
      GO SUB 2200: GO SUB 2300:
      GO SUB 2400
1099 GO TO 1001
2102 IF pr1=i1 THEN  LET sa1=1
2112 IF pr1=6 THEN  LET sa1=0
2113 GO SUB 7103
2120 LET pr1=pr1+1-sa1*2
2161 PRINT AT prr1+sa1*2,3;"  "
2170 PRINT AT pr1,3; INK 4;
      BRIGHT 1;d$;AT pr1+1,3;e$;
     AT pr1+2,3;f$
2190 LET prr1=pr1
2199 RETURN
2202 IF pr2=i2 THEN  LET sa2=1
2212 IF pr2=6 THEN  LET sa2=0
2213 GO SUB 7203
2220 LET pr2=pr2+1-sa2*2
2261 PRINT AT prr2+sa2*2,11;"
     "
2270 PRINT AT pr2,11; INK 2;
      BRIGHT 1;d$;AT pr2+1,11;e$
     ;AT pr2+2,11;f$
2290 LET prr2=pr2
2299 RETURN
2302 IF pr3=i3 THEN  LET sa3=1
2312 IF pr3=6 THEN  LET sa3=0
2313 GO SUB 7303
2320 LET pr3=pr3+1-sa3*2
2361 PRINT AT prr3+sa3*2,19;"
     "
2370 PRINT AT pr3,19; INK 6;
      BRIGHT 1;d$;AT pr3+1,19;e$
     ;AT pr3+2,19;f$
2390 LET prr3=pr3
2399 RETURN
2402 IF pr4=i4 THEN  LET sa4=1
2412 IF pr4=6 THEN  LET sa4=0
2413 GO SUB 7403
2420 LET pr4=pr4+1-sa4*2
2461 PRINT AT prr4+sa4*2,27;"
     "
2470 PRINT AT pr4,27; INK 1;
      BRIGHT 1;d$;AT pr4+1,27;e$
     ;AT pr4+2,27;f$
2490 LET prr4=pr4
2499 RETURN
```

LOAD

Pallonzolo

```
5210 INK 2: BRIGHT 1
5211 LET xx=0: LET yy=167:
      GO SUB 5280: LET yy=152:
      GO SUB 5290: LET xx=18:
      GO SUB 5290: LET xx=33:
      GO SUB 5285: LET xx=36:
      LET yy=175: GO SUB 5280:
      LET xx=39: GO SUB 5280
5212 LET xx=42: LET yy=152:
      GO SUB 5290: LET xx=60:
      GO SUB 5285: LET xx=67:
      PLOT xx,yy: DRAW 0,12:
      DRAW -3,3: DRAW -1,0:
      DRAW -2,-2: LET xx=70:
      PLOT xx+7,yy: DRAW -7,0:
      DRAW 7,15: DRAW -7,0
5213 LET xx=80: GO SUB 5290:
      LET xx=98: LET yy=175:
      GO SUB 5280: LET xx=101:
      LET yy=152: GO SUB 5290
5279 INK 7: BRIGHT 0: RETURN
5280 PLOT xx,yy: DRAW 0,-23:
      RETURN
5285 PLOT xx,yy: DRAW 0,15:
      RETURN
5290 PLOT xx+6,yy: DRAW 3,0:
      DRAW 1,1: DRAW 2,0: DRAW 2
     ,2: DRAW 0,2: DRAW 1,1:
      DRAW 0,3: DRAW -1,1:
      DRAW 0,2: DRAW -2,2:
      DRAW -2,0: DRAW -1,1:
      DRAW -3,0
5291 DRAW -1,-1: DRAW -2,0:
      DRAW -2,-2: DRAW 0,-2:
      DRAW -1,-1: DRAW 0,-3:
      DRAW 1,-1: DRAW 0,-2:
      DRAW 2,-2: DRAW 2,0:
      RETURN
6051: LET sa1=0: LET sa2=0:
      LET sa3=0: LET sa4=0:
      DIM z(4)
6060 LET pr1=7: LET pr2=7: LET p
     r3=7: LET pr4=7: LET prr1=7
     : LET prr2=7: LET prr3=7:
      LET prr4=7
6061 FOR n=0 TO 31: PRINT AT 4,n
     ;"J";AT 21,n;"I": NEXT n:
      FOR n=3 TO 27 STEP 8:
```

```
     PRINT AT 4,n;"    ";AT 21,n
    ;"    ": NEXT n
6062 FOR v=1 TO 4
6069 FOR n=0 TO 3: LET xx=n*64+2
     4: LET yy=139: GO SUB 8210:
     LET yy=3: GO SUB 8110+10*n
     : NEXT n
6070 LET i1=17: LET i2=16: LET i
     3=15: LET i4=14
6099 RETURN
6810 PRINT AT CODE p$(43-pa*2)-4
     0,CODE p$(44-pa*2)-40;
     OVER 1; INK 3;" ": LET pa=
     pa-1
6820 IF pa=0 THEN  GO TO 6900
6849 RETURN
6860 LET sc=sc+1: LET s$=STR$ sc

6861 PRINT AT 2,32-LEN s$;sc
6865 IF sc=9999 THEN  GO TO 6900

6899 RETURN
6910 PAUSE 5: PAUSE 30: FOR n=-5
     TO -35 STEP -5: BEEP -n/50
     ,n: NEXT n
6980 FOR n=4 TO 21: PRINT AT n,0
     ; PAPER 0;a$: NEXT n
6981 IF sc>hi THEN  LET hi=sc:
     LET r$=STR$ hi: PRINT AT 0
     ,32-LEN r$;hi
6999 GO SUB 7000: GO SUB 600:
     GO TO 990
7010 LET sc=0
7020 INK 4: PLOT 124,175: DRAW 2
     ,0: DRAW 0,-25: DRAW 59,0:
     DRAW 0,25: DRAW 2,0:
     DRAW 0,-27: DRAW -63,0:
     DRAW 0,27: INK 7
7021 LET pa=21: FOR n=0 TO 2:
     PRINT AT 2-n,16; INK 5;"LL
     LLLLL": BEEP .1,n↑3: NEXT n

7022 IF hi=0 THEN  PRINT AT 0,25
     ;"HI 0000"
7023 PRINT AT 2,25;"SC 0000"
7099 RETURN
7103 IF (pr1=17 OR pr1=6) AND z1
     =1 THEN  BEEP .1,30:
     GO SUB 6850: RETURN
7104 IF z1=0 AND (pr1<>17
     AND pr1<>6) THEN
     RETURN
7110 PRINT AT pr1,3; INK 4;
     BRIGHT 1;g$;AT pr1+1,3;g$;
     AT pr1+2,3;g$: BEEP .2,-5:
     PRINT AT pr1,3;"    ";AT pr
```

LOAD

Pallonzolo

```
     1+1,3;"    ";AT pr1+2,3;"
     ": GO SUB 6800: RETURN
7203 IF (pr2=16 OR pr2=6) AND z2
     =1 THEN  BEEP .1,30:
     GO SUB 6850: RETURN
7204 IF z2=0 AND (pr2<>16
     AND pr2<>6) THEN
     RETURN
7210 PRINT AT pr2,11; INK 2;
     BRIGHT 1;g$;AT pr2+1,11;g$
     ;AT pr2+2,11;g$: BEEP .2,-1
     0: PRINT AT pr2,11;"    ";
     AT pr2+1,11;"    ";AT pr2+2,
     11;"    ": GO SUB 6800:
     RETURN
7303 IF (pr3=15 OR pr3=6) AND z3
     =1 THEN  BEEP .1,30:
     GO SUB 6850: RETURN
7304 IF z3=0 AND (pr3<>15
     AND pr3<>6) THEN
     RETURN
7310 PRINT AT pr3,19; INK 6;
     BRIGHT 1;g$;AT pr3+1,19;g$
     ;AT pr3+2,19;g$: BEEP .2,-1
     5: PRINT AT pr3,19;"    ";
     AT pr3+1,19;"    ";AT pr3+2,
     19;"    ": GO SUB 6800:
     RETURN
7403 IF (pr4=14 OR pr4=6) AND z4
     =1 THEN  BEEP .1,30:
     GO SUB 6850: RETURN
7404 IF z4=0 AND (pr4<>14
     AND pr4<>6) THEN
     RETURN
7410 PRINT AT pr4,27; INK 1;
     BRIGHT 1;g$;AT pr4+1,27;g$
     ;AT pr4+2,27;g$: BEEP .2,-2
     0: PRINT AT pr4,27;"    ";
     AT pr4+1,27;"    ";AT pr4+2,
     27;"    ": GO SUB 6800:
     RETURN
8010 DATA 0,1,7,15,28,59,63,124,
     126,115,247,247,239,239,127
     ,127,127,63,63,31,15,7,1,0
8011 DATA 60,255,255,255,127,239
     ,159,127,255,255,255,255,25
     5,255,255,60
8012 DATA 0,128,224,240,248,252,
     252,254,254,254,255,255,255
```

```
     ,255,254,254,254,252,252,24
     8,240,224,128,0
8013 DATA 8,8,20,36,164,75,6,0,0
     ,96,210,37,36,40,16,16,8,65
     ,66,40,144,58,33,68,0,24,60
     ,110,126,60,24,0
8020 LET i=6: RESTORE 8010:
      FOR n=65368 TO 65399:
      READ k: POKE n,k: LET i=8-
     i: PRINT AT 11,8; OVER 1;
      BRIGHT 1; INK i;"
            ": BEEP .06,i*8:
      NEXT n
8049 FOR n=65400 TO 65463:
      READ k: POKE n,k: NEXT n:
      PRINT AT 11,8;"
          ": GO SUB 5200:
      RETURN
8070 LET a$="
                    ": LET b$="
     {G5}{9G3}              {9G3}
     {SG5}{G1}{4G3}
              {4G3}{G2}"
8071 LET c$="FERMA  IL NASTRO":
      LET d$="ADF": LET e$="B
     {SG8}G": LET f$="CEH":
      LET g$="KKK"
8072 LET p$="(8(9(:(;(<(=(>)8)9)
     :);)<)=)>*8*9*:*;*<*=*>"
8073 LET y$="SEI IN UNA GROTTA P
     IENA DISTALATTITI E STALAGM
     ITI CHE,SE URTATE DA GROSSI
      PALLONZOLISALTELLANTI, POS
     SONO FARLIESPLODERE ROVINAN
     DOLI.DEVI EVITARE CHE QUEST
     O ACCADAAZIONANDO GLI APPOS
     ITI COMANDIUN ATTIMO PRIMA
     CHE I PALLONZOLI"
8074 LET t$="TOCCHINO LE PARETI
     DELLA GROTTA.OGNI VOLTA CHE
      CI RIESCI FAIUN PUNTO.  I
     PALLONZOLI SONO 21.ATTENTO
     A NON PERDERE LA CALMA!!"
8099 RETURN
8110 PLOT xx,yy: DRAW 1,1:
      DRAW 1,0: DRAW 1,1: DRAW 0
     ,2: DRAW 1,1: DRAW 0,2:
      DRAW 1,1: DRAW 0,1: DRAW 1
     ,1: DRAW 0,1: DRAW 0,-1:
      DRAW 2,-2: DRAW 0,-2:
      DRAW 3,-3: DRAW 1,0:
      DRAW 1,1: DRAW 1,0
8111 DRAW 1,1: DRAW 1,0: DRAW 1,
     1: DRAW 0,1: DRAW 0,-1:
      DRAW 2,-2: DRAW 0,-2:
      DRAW 2,-2: DRAW 1,0
```

LOAD

Pallonzolo

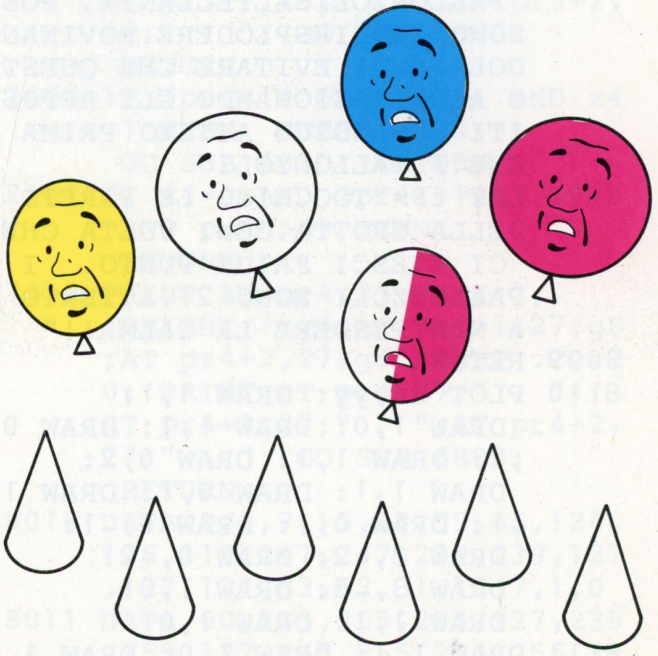
```
8112 PLOT xx+7,yy+1: DRAW 1,0:
      DRAW 1,1: PLOT xx+14,yy:
      DRAW 2,1: DRAW 1,0: DRAW 1
     ,1
8115 RETURN
8120 PLOT xx,yy: DRAW 1,1:
      DRAW 0,1: DRAW 1,0: DRAW 2
     ,2: DRAW 0,3: DRAW 1,1:
      DRAW 0,2: DRAW 3,2: DRAW 0
     ,1: DRAW 2,2: DRAW 1,1
8121 DRAW 0,3: DRAW 1,-1: DRAW 0
     ,-1: DRAW 2,-2: DRAW 0,-1:
      DRAW 2,-2: DRAW 0,-2:
      DRAW 1,-1: DRAW 0,-2:
      DRAW 1,-2: DRAW 0,-1:
      DRAW 1,-1: DRAW 0,-2:
      DRAW 1,-1
8122 DRAW 3,-1: PLOT xx+10,yy:
      DRAW 1,1: DRAW 1,0: DRAW 1
     ,1: DRAW 2,2: PLOT xx+9,yy+
     6: DRAW 0,3: DRAW 1,2:
      DRAW 1,1
8125 RETURN
8130 PLOT xx,yy: DRAW 1,1:
      DRAW 0,2: DRAW 1,1: DRAW 0
     ,1: DRAW 1,1: DRAW 1,0:
      DRAW 2,1: DRAW 1,3: DRAW 2
     ,3: DRAW 1,2: DRAW 1,1:
      DRAW 1,1: DRAW 0,1: DRAW 0
     ,-1
8131 DRAW 0,2: DRAW 1,1: DRAW 0,
     2: DRAW 1,3: DRAW 1,-3:
      DRAW 2,-1: DRAW 1,-5:
      DRAW 1,-2: DRAW 0,-3:
      DRAW 2,-5: DRAW 1,-4:
      DRAW 1,-1
8132 PLOT xx+11,yy+8: DRAW 1,1:
      DRAW 3,1: PLOT xx+12,yy+1:
      DRAW 3,2: DRAW 1,1: DRAW 1
     ,0: DRAW 1,1: PLOT xx+15,yy
     : DRAW 2,0: DRAW 1,1:
      DRAW 1,0
8135 RETURN
8140 PLOT xx,yy: DRAW 2,2:
      DRAW 0,2: DRAW 1,1: DRAW 0
     ,1: DRAW 1,1: DRAW 0,4:
      DRAW 3,3: DRAW 0,2: DRAW 1
     ,1: DRAW 0,2: DRAW 1,1:
```

```
      DRAW 0,2: DRAW 0,7: DRAW 1
     ,1: DRAW 0,4: DRAW 0,-4
8141 DRAW 1,-1: DRAW 0,-3:
      DRAW 1,-1: DRAW 0,-8:
      DRAW 1,-1: DRAW 0,-2:
      DRAW 2,-2: DRAW 2,2:
      DRAW 1,-1: DRAW 0,-3:
      DRAW 1,-1: DRAW 0,-2:
      DRAW 1,-1: DRAW 0,-2:
      DRAW 1,-1: DRAW 0,-2:
      DRAW 1,-1
8142 PLOT xx+15,yy+3: DRAW 1,1:
      DRAW -1,1: DRAW 2,1:
      DRAW -1,1: PLOT xx+7,yy+12
     : DRAW 1,1: DRAW -1,1:
      DRAW 1,2
8145 RETURN
8210 PLOT xx,yy: DRAW 1,-1:
      DRAW 3,0: DRAW 2,-2:
      DRAW 3,-4: DRAW 0,-2:
      DRAW 1,-1: DRAW 0,-1:
      DRAW 2,2: DRAW 0,1: DRAW 1
     ,1: DRAW 0,4: DRAW 1,3:
      DRAW 1,2: DRAW 1,1: DRAW 1
     ,0: DRAW 1,-1: DRAW 2,-3:
      DRAW 1,-1: DRAW 1,-3
8211 DRAW 0,-1: DRAW 1,7: DRAW 1
     ,0: PLOT xx+10,yy: DRAW 1,1
     : DRAW 1,1: DRAW 0,2
8299 RETURN
8510 PRINT #1;AT 0,0; PAPER 0;
      INK 1;b$;AT 0,11; INK 6;"P
     ALLONZOLO";AT 1,6; INK 7;"©
      1984 by DISCOVOGUE"
8519 POKE 23606,16: POKE 23607,1
     64
8520 PRINT AT 11,8; INK 6;
      BRIGHT 1;c$
8599 RETURN
9000  REM  LINEE DI GOSUB

9160 RETURN
9161 LET z1=1: LET z2=1: LET z3=
     1: LET z4=1: RETURN
9162 RETURN
9163 LET z2=1: LET z3=1: LET z4=
     1: RETURN
9164 RETURN
9165 LET z1=1: LET z3=1: LET z4=
     1: RETURN
9166 RETURN
9167 LET z3=1: LET z4=1:
      RETURN
9168 RETURN
9169 LET z1=1: LET z2=1: LET z4=
     1: RETURN
```

LOAD

Pallonzolo

```
9170 RETURN
9171 LET z2=1: LET z4=1:
      RETURN
9172 RETURN
9173 LET z1=1: LET z4=1:
      RETURN
9174 RETURN
9175 LET z4=1: RETURN
9176 RETURN
9177 LET z1=1: LET z2=1: LET z3=
     1: RETURN
9178 RETURN
9179 LET z2=1: LET z3=1:
      RETURN
9180 RETURN
9181 LET z1=1: LET z3=1:
      RETURN
9182 RETURN
9183 LET z3=1: RETURN
9184 RETURN
9185 LET z1=1: LET z2=1:
      RETURN
9186 RETURN
9187 LET z2=1: RETURN
9188 RETURN
```



AFFIDA I TUOI DATI A UN SUPPORTO SICURO

Come editori di software, abbiamo sentito l'esigenza di utilizzare, per la produzione dei nostri programmi, un supporto particolarmente affidabile. Dopo severi ed accurati test abbiamo operato la scelta. Siamo lieti di proporlo con il nostro marchio a chiunque desideri lavorare con la nostra stessa tranquillità. Floppy disk da 5'' 1/4, singola faccia, doppia densità, in confezione da 10 dischetti. Ordine minimo 10 dischetti. Ordini superiori solo multipli di 10 secondo la seguente scala di prezzi

- 10 dischetti	L. 5.000 cad.
- da 20 a 50 dischetti	L. 4.700 cad.
- da 60 a 100 dischetti	L. 4.400 cad.
- da 110 dischetti e oltre	L. 3.900 cad.

I prezzi sono comprensivi di IVA e spese di spedizione.

Per ordinare ritagliate e spedite il tagliando sotto riportato a J. *soft* - via Rosellini, 12 - 20124 Milano
Tel. 02/6888228 - 683797 - 6880841 - 6880842 - 6880843

CEDOLA DI ORDINAZIONE OFFERTA DISCHETTI

Da compilare e spedire in busta chiusa a
J. *soft* - via Rosellini, 12 - 20124 Milano
Tel. 02/6888228 - 683797 - 6880841 - 6880842 - 6880843

Ordino i seguenti dischetti, in confezione da 10 pezzi cad., per un importo totale di L. ..... IVA e spese di spedizione incluse.

- ☐ N. ..... dischetti (minimo 10 e multipli di 10)
- ☐ Contanti allegati
- ☐ Assegno allegato n° .....
- ☐ Ho spedito l'importo a mezzo vaglia postale
- ☐ Ho versato l'importo sul CCP n° 19445204 intestato a J. *soft* - Milano
- ☐ Pagherò in contrassegno al postino al ricevimento dei dischetti

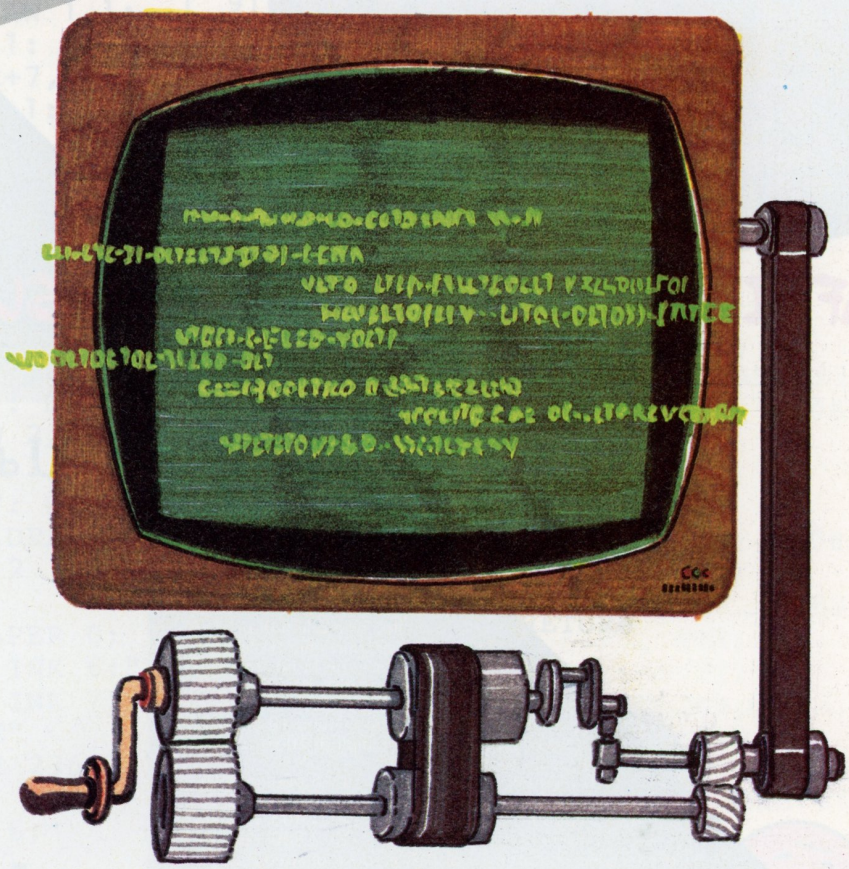
Nome .....
Cognome .....
Via .....
CAP ..... Città ..... Prov. .....
Se richiesta fattura - codice fiscale .....
Data ..... Firma .....

Offerta valida solo per l'Italia.

Scroll control

di **Emilio Roncoroni**

Una corta, ma potente routine in linguaggio macchina per ottenere sorprendenti effetti grafici



L'utility di seguito presentata permette di scrollare, pixel per pixel ad anello, le righe dalla 0 alla 23, a sinistra o a destra, indipendentemente fra di loro e a diverse velocità di spostamento.
La realizzazione della routine avviene in tre passaggi:
1) digitare il programma BASIC e salvarlo su cassetta;
2) decidere quali righe dello schermo video si vogliono "scrollare", scegliere quali si dovranno spostare a destra e quali a sinistra ed infine a quale velocità relativa si dovranno spostare le une rispetto alle altre;
3) se ad esempio avete scelto di "scrollare" le righe:
— 04 a destra velocità 3
— 08 a destra velocità 2
— 10 a sinistra velocità 1
inserirete le seguenti istruzioni nel programma BASIC:

```
10 LET a$ = "04d308d210s1"
20 LET b = 255
30 GOSUB 9500
40 GOSUB 9600
```

eseguite un RUN.

Avete ora a disposizione una routine in L/M di "scroll" delle righe scelte;per poterla salvare eseguite un GO TO 9980.
L'istruzione 10 abbisogna di una breve spiegazione: la variabile alfanumerica a$ deve essere inizializzata nel seguente modo: 2 byte per la linea da scrollare (sempre due byte anche se si scelgono le righe 1/2/3 ecc., e cioè inserire 01/02/03 ecc.), 1 byte per la direzione di spostamento (d = destra, s = sinistra), 1 byte da 1 a 9 per la velocità relativa di spostamento e di seguito le altre linee da scrollare.
Se volete scrollare contemporaneamente tutto lo schermo video, dovrete impostare a$ nel seguente modo: LET a$ = "24d5", il che vi permetterà di

Descrizione del programma

Linee	Commento
BASIC	
9500-9540	Caricano le due routine in L/M
9610	Memorizza in RAM la variabile b (numero di cicli)
9625	Verifica se è stato scelto lo "scroll" totale del video; se si, GOSUB 9930
9640	Inizio del ciclo di creazione del programma in L/M
9650-9680	Con lo "slicing" delle variabili a$ e b$ si ottengono le variabili c$ (numero di riga), d$ (destra/sinistra), e$ (velocità relativa)
9690-9700	Verifiche di validità di d$ e di e$
9730-9880	Determinano la variabile f, ovvero la locazione di inizio della riga analizzata nel Display File
9810-9880	Creano il programma in L/M
9894-9899	Creano la coda del programma in L/M
9930-9977	Subroutine per la creazione della variabile a$ per lo scroll di tutte le righe del video

scrollare tutto lo schermo a destra a velocità 5; naturalmente, potrete inserire s in sostituzione di d e lo schermo scrollerà a sinistra. La velocità potrà essere impostata a piacere da 1 a 9.
L'istruzione 20 è la definizione della variabile b, ovvero il numero di cicli che vorrete vengano eseguiti dalla routine in L/M (il numero massimo consentito è 255).
Quando vorrete utilizzare la routine non dovrete far altro che caricarla con LOAD "" CODE e con RANDOMIZE USR 60066 essa verrà eseguita.

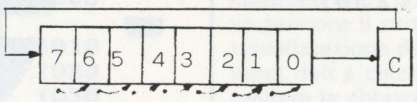
Esempio:
LOAD ""CODE
10 FOR u=0 TO 21
20 FOR v=0 TO 31
30 PRINT AT u,v,"X"
40 NEXT v
50 NEXT u
60 FOR s=o TO 21
70 PRINT s
80 NEXT s
90 RANDOMIZE USR 60066
eseguite un RUN
Lo schermo verrà riempito di "X", le righe verranno numerate da 0 a 21 e le righe 4, 8 e 10 scrolleranno nel modo da voi voluto (vedi punto 3).
Esiste la possibilità di modificare in qualsiasi momento il numero dei cicli della routine, "pokando" opportunamente nella locazione 60067 (date POKE 60067,1 e le righe interessate si sposteranno di un pixel).
Naturalmente potrete creare diverse routine di "scroll" ripetendo le fasi 2 e 3 (preventivamente andrà caricato il programma BASIC salvato nella fase 1).
Nei listati 1 e 2 potete vederne un esempio.
La routine in L/M sinistra esegue le stesse operazioni con la sola differenza di ruotare i bit a sinistra con l'istruzione RLA.

Linguaggio macchina (L/M)
(Routine destra)

Loc. Code	mnemonico	note
EA6F 0E08	LD C, 08	carica nel registro C il numero 8 decimale
EA71 E5	PUSH HL	salva nello stack il doppio registro HL, il cui contenuto è stato caricato dal programma principale in l/m e che corrisponde alla prima locazione della riga Display File
EA72 111F00	LD DE,001F	carica il numero 31 decimale nel doppio registro DE
EA75 19	ADD HL,DE	somma i contenuti dei registri DE e HL
EA76 7E	LD A,(HL)	carica il contenuto della locazione puntata dal doppio registro HL nel registro A
EA77 ED52	SBC HL,DE	sottrae dal registro HL il valore contenuto nel doppio registro DE
EA79 1F	RRA	ruota a destra i bit del registro A
EA7A 0620	LD B,20	carica il numero 32 decimale nel registro B e prepara un loop di 32 cicli
EA7C 7E	LD A,(HL)	carica il contenuto della locazione puntata dal doppio registro HL nel registro A
EA7D 1F	RRA	ruota a destra
EA7E 77	LD (HL),A	dopo aver ruotato i bit del registro A li rimette nel Display File
EA7F 23	INC HL	incrementa di 1 il doppio registro HL
EA80 10FA	DJNZ EA7C	decrementa di 1 il registro B e test del flag Z: se è uguale a 0, prosegue (fine loop), altrimenti salta all'inizio del loop (EA7C)
EA82 E1	POP HL	riprende dallo stack il valore originario del doppio registro HL precedentemente salvato
EA83 24	INC H	incrementa di 1 il registro H, ovvero il byte più significativo del doppio registro HL; è come sommare 256 al doppio registro HL
EA84 0D	DEC C	decrementa il registro C
EA85 20EA	JR NZ,EA71	test del flag Z: se è 0, prosegue (vengono così spostati tutti gli 8 x 32 pixel della riga), altrimenti salta al loop (EA71)
EA87 C9	RET	ritorna al BASIC

Esempio pratico di come lavora la routine

Premesso che ogni riga del video è organizzata in 8 x 32 pixel in orizzontale x 8 pixel in verticale = 2048 pixel, la routine sposta con un loop (il cui ciclo è uguale a 8) 8 x 32 pixel orizzontali in ciascun ciclo.
Ammettendo di voler far ruotare ad anello a destra ciò che è visualizzato nella riga 0, la routine eseguirà le seguenti operazioni:
1. prenderà il primo degli 8 byte che formano il carattere della 32a colonna, gli farà ruotare i bit a destra, ovvero



memorizzando così l'ultimo pixel della 32a colonna nel carry;
2. inizierà un loop di 32 cicli a partire dalla locazione 16384 (locazione di inizio della riga 0 nel Display File), ruoterà i bit a destra come prima descritto e metterà il contenuto del carry, precedentemente memorizzato, nel bit 7 del byte corrente, e così sino a completare il loop di 32 cicli;
3. ripeterà il loop di cui il punto 2 per 8 volte, analizzando così tutti i 256 byte che compongono i caratteri "mostrati" sulla riga.

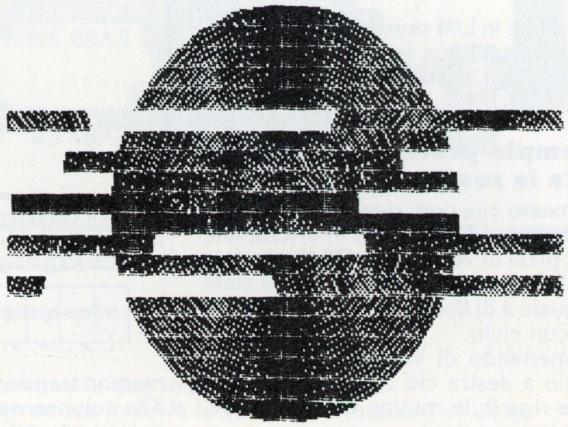
Scroll control versione per ZX Spectrum 16K/48K

programma 1 - contiene scroll mostrato nella foto

```
10 LET a$="07d109s211d313s415d
   5"
20 LET b=255
30 GO SUB 9500
40 GO SUB 9600
9490 STOP
9495 REM  © E. RONCORONI
9500 FOR k=1 TO 52
9510 READ h
9520 POKE (60014+k),h
9530 NEXT k
9540 RETURN
```

Scroll control

```
9600 REM  POKE 60067,(b-256*
     INT (b/256))
9610 REM  POKE 60068,(INT (b/256
     ))
9611 POKE 60067,b: POKE 60068,0
9620 LET n=60069
9625 LET c$=a$(1 TO 2): IF c$>="
     24" THEN  GO SUB 9930
9630 LET c=LEN a$
9640 FOR x=1 TO c STEP 4
9650 LET b$=a$(x TO (x+3))
9660 LET c$=b$(1 TO 2)
9670 LET d$=b$(3 TO 3)
9680 LET e$=b$(4 TO 4)
9690 IF d$<>"d" AND d$<>"s"
      THEN  LET d$="d"
9700 IF e$="0" THEN  LET e$="1"
9710 LET rip=VAL e$
9720 LET sco=VAL c$
9730 IF sco>=0 AND sco<=7 THEN
      LET f=16384
9740 IF sco>=8 AND sco<=15
      THEN  LET f=18432: LET sco
     =sco-8
9750 IF sco>=16 AND sco<=23
      THEN  LET f=20480: LET sco
     =sco-16
9760 LET f=f+(sco*32)
9770 IF d$="d" THEN  LET f=f+0:
      LET w=111
9780 IF d$="s" THEN  LET f=f+31:
      LET w=136
9790 LET z=f-256*INT (f/256)
9800 LET y=INT (f/256)
9810 FOR q=1 TO rip
9820 RESTORE 9996
9830 FOR g=1 TO 8
9840 READ m
9850 POKE n,m
9860 LET n=n+1
9870 NEXT g
9880 NEXT q
9885 IF n>62000 THEN  GO TO 9894
9890 NEXT x
9894 RESTORE 9997
9895 LET n=n-1
9896 FOR t=1 TO 5
9897 READ o
9898 POKE (n+t),o
9899 NEXT t
9920 RETURN
9930 LET f$=a$(3 TO 4): LET a$="
     "
9940 FOR j=0 TO 23
9950 IF j=0 THEN  LET a$=a$+"00"
     : GO TO 9970
9960 IF j<10 THEN  LET a$=a$+"0"

9965 LET a$=a$+STR$ j
9970 LET a$=a$+f$
9975 NEXT j
9977 RETURN
9980 SAVE "spt"CODE 60015,(n+8-6
     0015)
9981 STOP
9995 DATA 14,8,229,17,31,0,25,12
     6,237,82,31,6,32,126,31,119
     ,35,16,250,225,36,13,32,234
     ,201,14,8,175,229,17,31,0,2
     37,82,126,25,23,6,32,126,23
     ,119,43,16,250,225,36,13,32
     ,233,201,38
9996 DATA 229,33,z,y,205,w,234,2
     25
9997 DATA 37,194,165,234,201
```



Esempio di composizione grafica programma 2 - vedi foto

```
10 FOR i=1 TO 80
20 CIRCLE 127,87,i
30 NEXT i
```

Data base hashing

di **Antonio Motta**

Load 'DATA BASE HASH

Un metodo di ricerca veloce applicato ad un home computer

Uno degli usi più comuni dei calcolatori è la memorizzazione di dati in tabelle. Per un utilizzo razionale degli archivi elettronici sono state ideate numerose tecniche di ricerca. Dalla semplice ricerca sequenziale, che non richiede nessun tipo di ordinamento dei dati all'interno della tabella, si è passati alla ricerca binaria. Questa tecnica, che richiede l'ordinamento dei nomi memorizzati, è più veloce della ricerca sequenziale (1). La ricerca binaria può essere ulteriormente migliorata, facendo in modo che i dati consultati più spesso vengano spostati all'inizio della tabella; in questo modo si riduce ancora il tempo medio di ricerca (2). Tutti questi metodi, però, hanno una caratteristica in comune: il tempo di ricerca di un dato è proporzionale alla lunghezza della tabella. I metodi considerati finora, quindi, non sono i più veloci nella ricerca in tabelle molto grandi; esiste però un'alternativa nel metodo hashing.

Il metodo hashing

L'idea consiste nel considerare ogni dato per le sue caratteristiche individuali, anziché ordinarlo rispetto al contesto in cui si trova.
In pratica il dato viene scomposto nelle sue lettere e ad ogni lettera viene assegnato un numero diverso (p.esempio A = 1...Z = 26).
La somma dei numeri ottenuti è il codice hashing del dato, che fa da puntatore nel vettore dati. Per esempio, per memorizzare il nome ROSSI si assegna ad ogni lettera il numero che individua la sua posizione nell'alfabeto: R = 18, O = 15, S = 19, I = 9; si sommano i numeri ottenuti: 18 + 15 + 19 + 19 + 9 = 80 e si inserisce il nome ROSSI nel vettore dati in posizione 80. In fase di ricerca si ripete il calcolo del codice hashing di ROSSI e lo si va a cercare in posizione 80, ritrovandolo al primo confronto.
Una prima complicazione nasce quando il codice hashing è maggiore dell'indice massimo del vettore. Nel nostro esempio, se il vettore ha 50 posizioni, si divide il codice hashing (80) per 50, e il resto della divisione (30) diventa l'indice del nome ROSSI nel vettore. In altri termini: codice hashing = 80 modulo 50. Il secondo problema nasce quando si verifica una collisione, cioè quando due o più dati hanno lo stesso codice.
Questo problema si può risolvere in almeno due modi. Il primo, e più semplice, dei due consiste nell'inserire il secondo dato nella prima posizione libera precedente al primo. Per esempio, volendo aggiungere al nostro vettore il nome SORSI (che ovviamente ha lo stesso codice di ROSSI), questo, essendo occupata la posizione 30, sarà memorizzato in posizione 29. Il secondo metodo invece realizza l'ordinamento alfabetico dei dati che hanno lo stesso

Descrizione del programma

Linee	Commento
20-60	definizione record
70-80	calcolo dei record memorizzabili. Le costanti 5 e 36460 (indirizzo d'inizio zona variabili) servono per l'ottimizzazione del calcolo della memoria disponibile e sono state trovate per tentativi. Questi valori sono validi per la versione da 48 K; chi ha la versione da 16 K dovrà adattare queste due costanti alla memoria disponibile (questa è l'unica modifica da fare per adattare il programma alla versione non espansa)
90-100	visualizzazione del numero di record memorizzabili e richiesta di convalida
110-130	stampa menu e attesa comandi
140-160	riconoscimento del comando
200-240	subroutine per il calcolo del codice hashing. L'istruzione N° 3 della linea 200 identifica la fine del dato nel caso che questo sia più corto della stringa già dimensionata.
1000	controlla se c'è posto per un altro record; in caso negativo provoca il ritorno al menu
1010	visualizzazione del numero di record ancora disponibili
1020	input dati e calcolo del codice hashing
1030	realizza la chiusura ad anello dell'archivio
1040	controlla se l'indirizzo hashing è libero e in caso affermativo provoca un salto all'istruzione di memorizzazione del record
1050	controlla se il dato in ingresso è alfabeticamente precedente al dato già presente all'indirizzo hashing; in caso affermativo passa al controllo dell'indirizzo precedente
1060	controlla se c'è già un record memorizzato con la stessa chiave nel campo N° 1; in caso affermativo realizza l'ordinamento alfabetico in base al contenuto dei campi N° 2
1070-1080	nel caso che il dato in ingresso sia alfabeticamente seguente al dato già memorizzato all'indirizzo hashing, i 2 record si scambiano tra loro
1090	memorizza il record in ingresso al suo indirizzo hashing
1100	decrementa di 1 il numero di record disponibili e provoca il ritorno al menu

Cancellazione (Figura 2)

Questa è la fase più lunga da eseguire, perché ogni volta che si cancella un record bisogna ricontrollare tutto l'archivio alla ricerca di eventuali record da riordinare. Di conseguenza la durata della cancellazione è proporzionale alla lunghezza dell'archivio. L'algoritmo proposto è senz'altro migliorabile: in particolare vedere la linea 2100.

2000-2010	input dato e calcolo del codice hashing
2020-2030	fanno proseguire la ricerca del dato da cancellare
2040	provoca il ritorno al menu in caso di assenza del dato impostato
2050-2060	controllano se ci sono più record con la stessa chiave nel campo N° 1
2070	cancella il record
2080-2090	controllano l'analisi dell'archivio e provocano il ritorno al menu quando tutti i record sono stati analizzati
2100	controlla se ci sono record con codice hashing maggiore o uguale a quello del record cancellato; in caso affermativo li riordina. La seconda condizione analizzata dalla linea è un modo empirico per riordinare i dati con codice hashing molto basso e spostati negli indirizzi più alti del vettore: infatti, nel caso che questi dati siano seguiti da un altro dato memorizzato al suo indirizzo hashing (cioè con codice hashing maggiore del loro) in assenza di questa condizione non verrebbero spostati agli eventuali posti liberati all'inizio del vettore
2110	proseguimento dell'analisi dell'archivio
2120-2170	visualizzazione di tutti i record aventi la chiave data e input del numero del record da cancellare

Ricerca (Figura 3)

3000-3010	input chiave di ricerca e calcolo del codice hashing
3020	realizza la chiusura ad anello dell'archivio
3030	fa proseguire la ricerca nel caso che il dato cercato sia alfabeticamente precedente a quello trovato
3040	conclude la ricerca e provoca il ritorno al menu
3050-3080	visualizzano e stampano il record cercato
3090	fa proseguire la ricerca

codice hashing. Nel nostro caso ROSSI sarà spostato in posizione 29 e SORSI andrà in posizione 30.

I vantaggi dell'ordinamento hashing-alfabetico si avvertono in caso di assenza della chiave cercata, perché in questo caso la ricerca termina quando viene trovato un dato alfabeticamente precedente a quello cercato.

In un normale archivio hashing, invece, la ricerca prosegue fino al ritrovamento di un record vuoto. Va da sé che la cancellazione di un nome da un archivio ordinato hashing-alfabeticamente deve essere seguita dal riordinamento dei nomi rimanenti. Proseguendo nel nostro esempio, dopo aver cancellato il nome SORSI la posizione 30 rimane libera; successivamente, se si va a cercare ROSSI nella sua posizione hashing, cioè 30, essendo questa libera la ricerca si concluderà immediatamente con esito negativo. Quindi è fondamentale ordinare i dati anche dopo una cancellazione. Due ultime osservazioni: la prima è che le ricerche e gli ordinamenti passano dalla prima all'ultima posizione del vettore, che quindi viene trattato come una struttura chiusa ad anello. La seconda osservazione è che nel vettore deve esserci almeno una posizione libera, altrimenti la ricerca di un nome assente ed alfabeticamente inferiore a tutti i nomi memorizzati proseguirebbe all'infinito (2)

Bibliografia:

1) E. Mitchell, "Tecniche di ricerca", Personal Software N. 5, aprile 1983.

2) D.E. Knuth, "Gli algoritmi", Le Scienze N. 108, agosto 1977, ristampato in Le Scienze quaderni N. 14, marzo 1984.

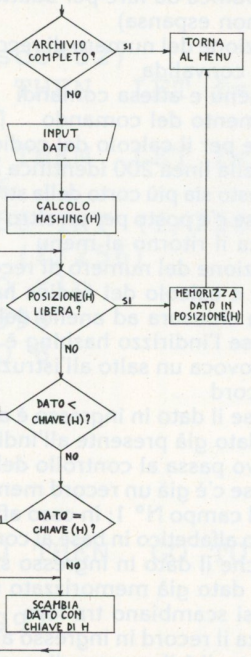


Fig. 1 - Flow-chart dell'aggiornamento

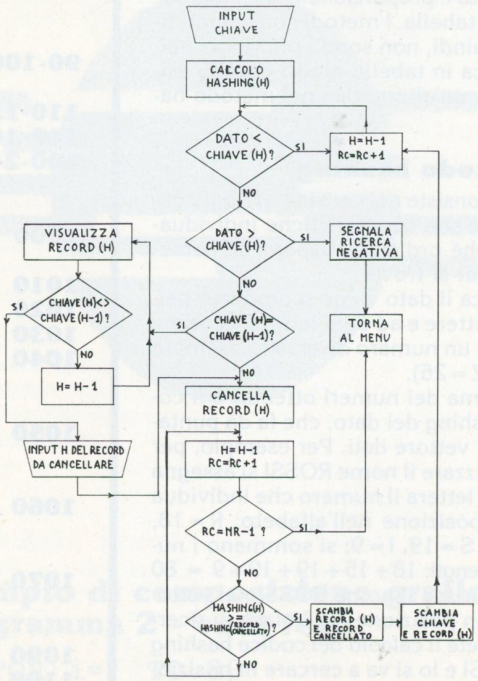


Fig. 2 - Flow-chart della cancellazione

Il programma

Il programma si usa in due fasi distinte; la prima permette di definire la struttura del record. In particolare bisogna decidere: 1) il numero di campi per record; 2) il nome di ciascun campo; 3) il numero di caratteri per campo. Una volta introdotti questi parametri il programma calcola il numero di record memorizzabili e visualizza il menu, segnalando così che si è passati alla fase di utilizzo dell'archivio.
A questo punto si può cominciare il lavoro di memorizzazione vero e proprio. Il programma non comprende un'opzione di modifica dei record; ho preferito non metterla per lasciare più memoria per i dati. D'altronde è possibile cancellare un record e memorizzarlo nella versione modificata. Per salvare i dati si usa l'opzione N. 5 del menu, che fa un SAVE del programma e quindi anche delle variabili memorizzate. Quando verrà caricata questa seconda registrazione il programma partirà direttamente dal menu, saltando la fase di definizione del record.
L'ordinamento avviene in base alla chiave memorizzata nel 1° campo.

Lista (Figura 4)

L'opzione Lista prevede due possibilità: si possono listare indistintamente tutte le chiavi memorizzate nei campi N° 1, oppure si possono listare tutti i record che contengono la chiave indicata nel campo desiderato. Per esempio, in un archivio bibliografico è possibile listare tutti i record relativi ai libri di uno stesso Editore, o di uno stesso Autore.

4000	input delle condizioni di lista
4010-4030	lista totale
4040-4090	input per la lista condizionata
4100-4110	lista condizionata
4120	ritorno al menu

Descrizione delle variabili

nc	numero campi
lc	lunghezza campi
cr	numero di caratteri per record
nr	numero record
fre	numero record vuoti
h	codice hashing
cod	codice hashing dell'ultima chiave cancellata
s	memorizza il codice hashing durante gli spostamenti dei record
rc	numero di record controllati dopo una cancellazione
m	h-l; serve per controllare se ci sono più record con la stessa chiave nel campo N° 1

Matrici

c$(nc,10)	nomi dei campi
n$(nc,lc)	usata per l'input di un record
m$(nr,nc,lc)	memorizzazione dati
s$(nc,lc)	usata durante gli spostamenti dei record
c(nr)	memorizzazione codici hashing

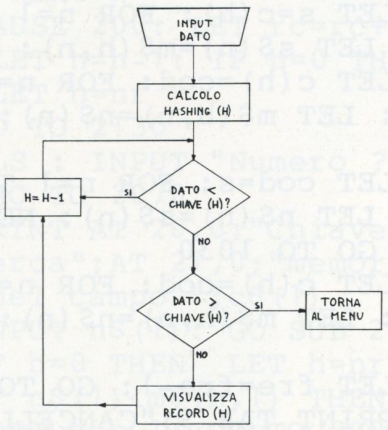


Fig. 3 - Flow-chart della ricerca

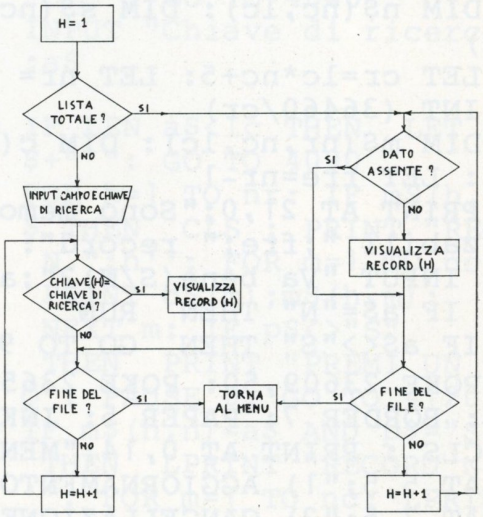
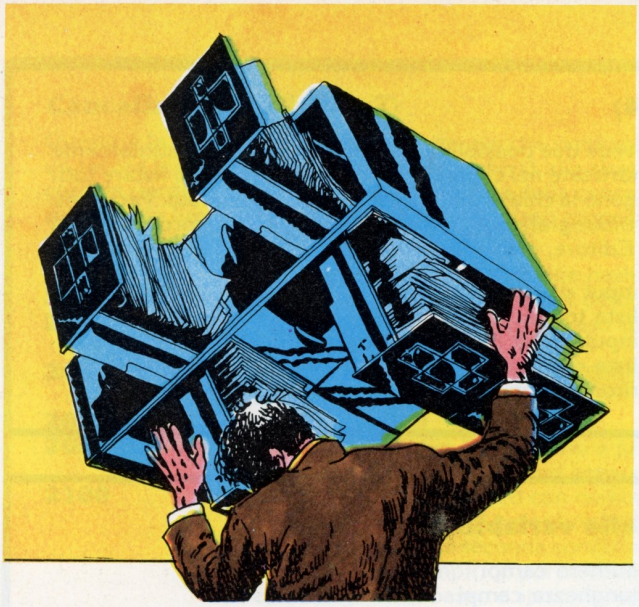


Fig. 4 - Flow-chart della lista



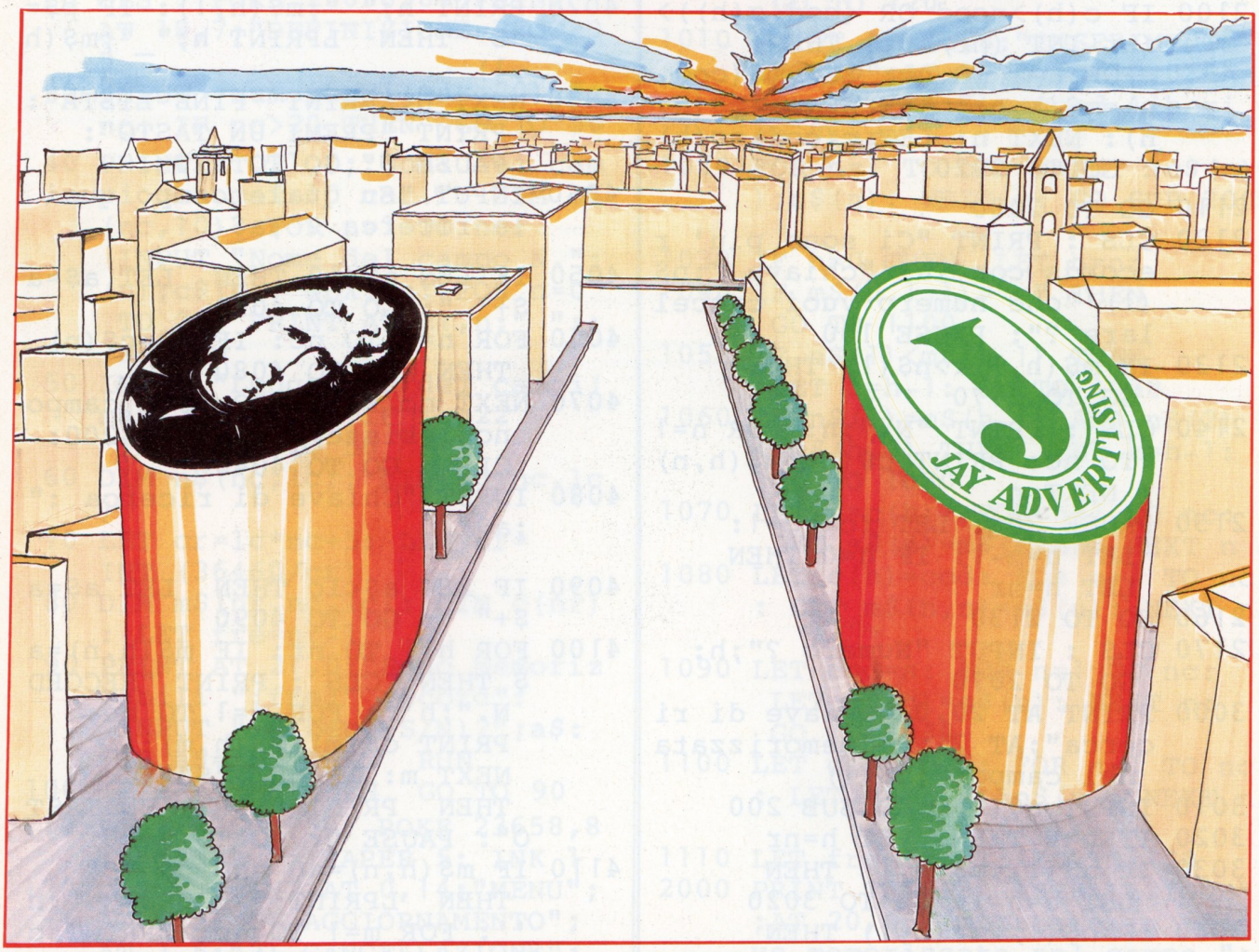
Data base hashing versione per ZX Spectrum 48K

```
20 CLS : POKE 23658,8: PRINT
   AT 18,7;"DEFINIZIONE RECORD
   "
30 INPUT "Num.campi(max 20):";
   nc: IF nc>20 THEN  GO TO 30
40 PRINT AT 21,0;"Nomi dei cam
   pi con max.10 car.": DIM c$
   (nc,10): FOR n=1 TO nc:
    INPUT "Nome del campo N.";
   (n);c$(n): NEXT n: FOR n=0
    TO 29: PRINT AT 21,n;" ";:
    NEXT n
50 INPUT "Lunghezza dei campi(
   max 22):";lc: IF lc>22
    THEN  GO TO 50
60 DIM n$(nc,lc): DIM s$(nc,lc
   )
70 LET cr=lc*nc+5: LET nr=
   INT (36460/cr)
80 DIM m$(nr,nc,lc): DIM c(nr)
   : LET fre=nr-1
90 PRINT AT 21,0;"Sono memoriz
   zabili ";fre;" record":
    INPUT "Va bene(S/N)?";a$:
    IF a$="N" THEN  RUN
100 IF a$<>"S" THEN  GO TO 90
110 POKE 23609,50: POKE 23658,8
    : BORDER 7: PAPER 5: INK 1
120 CLS : PRINT AT 0,14;"MENU";
    AT 5,5;"1) AGGIORNAMENTO";
    AT 7,5;"2) CANCELLAZIONE";
    AT 9,5;"3) RICERCA";AT 11,5
    ;"4) LISTA";AT 13,5;"5) REG
    ISTRAZIONE";AT 15,5;"6) FIN
    E"
130 PAUSE 0: LET a=VAL INKEY$:
     IF a<1 OR a>6 THEN
     GO TO 130
140 CLS : IF a=6 THEN  STOP
150 IF a=5 THEN  INPUT "Nome de
    l file: ";a$: SAVE a$
     LINE 110: GO TO 120
160 GO TO a*1000
200 LET h=0: FOR n=1 TO lc:
     IF CODE n$(1)(n TO n)=32
     THEN  LET n=lc: GO TO 220
210 LET h=h+CODE n$(1)(n TO n)-
    64
220 NEXT n
230 IF h>nr THEN  LET h=h-INT (
    h/nr)*nr
240 LET cod=h: RETURN
1000 IF fre=1 THEN  PRINT "ARCHI
     VIO COMPLETO": PAUSE 200:
      GO TO 120
1010 PRINT AT 15,8;"INSERIMENTO
     RECORD";AT 17,0;AT 18,0;"An
     cora ";fre;" record liberi
     su ";nr-1
1020 FOR n=1 TO nc: INPUT (c$(n)
     );n$(n): NEXT n: GO SUB 200
1030 IF h=0 THEN  LET h=nr
1040 IF m$(h,1,1)=" " THEN
      GO TO 1100
1050 IF n$(1)<m$(h,1) THEN
      LET h=h-1: GO TO 1030
1060 IF n$(1)=m$(h,1) AND n$(2)
     <=m$(h,2) THEN  LET h=h-1:
      GO TO 1030
1070 LET s=c(h): FOR n=1 TO nc:
      LET s$(n)=m$(h,n): NEXT n
1080 LET c(h)=cod: FOR n=1 TO nc
     : LET m$(h,n)=n$(n): NEXT n

1090 LET cod=s: FOR n=1 TO nc:
      LET n$(n)=s$(n): NEXT n:
      GO TO 1030
1100 LET c(h)=cod: FOR n=1 TO nc
     : LET m$(h,n)=n$(n): NEXT n

1110 LET fre=fre-1: GO TO 120
2000 PRINT TAB 9;"CANCELLAZIONE"
     ;AT 20,0;"Impostare la chia
     ve memorizzata nel campo ";
     c$(1)
2010 INPUT n$(1): GO SUB 200:
      LET rc=0
2020 IF h=0 THEN  LET h=nr
```

```
2030 IF n$(1)<m$(h,1) THEN
      LET h=h-1: LET rc=rc+1:
      GO TO 2020
2040 IF n$(1)>m$(h,1) THEN
      CLS : PRINT "RECORD SCONOS
     CIUTO": PAUSE 100: GO TO 12
     0
2050 LET m=h-1: IF m=0 THEN
      LET m=nr
2060 IF m$(m,1)=n$(1) AND m$(m,1
     ,1)<>" " THEN  GO TO 2120

2070 FOR n=1 TO nc: LET m$(h,n)=
     " ": NEXT n: LET c(h)=0:
      LET cod=h
2080 LET h=h-1: LET rc=rc+1:
      IF h=0 THEN  LET h=nr
2090 IF rc=(nr-1) THEN  LET fre=
     fre+1: GO TO 120
2100 IF c(h)>=cod OR (cod-c(h))>
     (nr-INT (nr/10)) THEN
      FOR n=1 TO nc: LET m$(cod,
     n)=m$(h,n): LET n$(n)=m$(h,
     n): NEXT n: LET c(cod)=c(h)
     : GO TO 2020
2110 GO TO 2080
2120 CLS : PRINT "Ci sono piu' r
     ecords con la"'"chiave ";n$
     (1)'"che numero vuoi cancel
     lare ?": PAUSE 100
2130 IF m$(h,1)<>n$(1) THEN
      GO TO 2170
2140 CLS : PRINT "N.";h: FOR n=1
      TO nc: PRINT c$(n);m$(h,n)
     : NEXT n
2150 PAUSE 200: LET rc=rc+1:
      LET h=h-1: IF h=0 THEN
      LET h=nr
2160 GO TO 2130
2170 CLS : INPUT "Numero ?";h:
      GO TO 2070
3000 PRINT AT 20,0;"Chiave di ri
     cerca";AT 21,0;"memorizzata
      nel campo ";c$(1)
3010 INPUT n$(1): GO SUB 200
3020 IF h=0 THEN  LET h=nr
3030 IF n$(1)<m$(h,1) THEN
      LET h=h-1: GO TO 3020
3040 IF n$(1)>m$(h,1) THEN
      CLS : PRINT "RICERCA TERMI
     NATA": PAUSE 100: GO TO 120

3050 CLS : PRINT "RECORD N.";h;"
          cod.hashing=";c(h)''
3060 FOR n=1 TO nc: PRINT c$(n);
     m$(h,n): NEXT n
3070 PRINT #0;"STAMPA (S/N)?":
      PAUSE 0
3080 IF INKEY$="S" THEN
      LPRINT "RECORD N.";h'':
      FOR n=1 TO nc: LPRINT c$(n
     );m$(h,n): NEXT n: LPRINT '
     ''

3090 INPUT "": LET h=h-1:
      GO TO 3020
4000 INPUT "Lista totale o condi
     zionata      (T/C) ?";a$:
      INPUT "STAMPA (S/N)?";p$:
      IF a$="C" THEN  GO TO 4040

4010 FOR h=1 TO nr: IF m$(h,1,1)
     =" " THEN  GO TO 4030
4020 PRINT h;"_";m$(h,1): IF p$=
     "S" THEN  LPRINT h;"_";m$(h
     ,1)
4030 NEXT h: PRINT "FINE LISTA":
      PRINT "PREMI UN TASTO":
      PAUSE 0: GO TO 120
4040 INPUT "Su quale campo vuoi
     la ricerca ?";a$

4050 IF LEN a$<10 THEN  LET a$=a
     $+" ": GO TO 4050
4060 FOR n=1 TO nc: IF a$=c$(n)
      THEN  GO TO 4080
4070 NEXT n: PRINT "Questo campo
      non esiste !": PAUSE 200:
      CLS : GO TO 4040
4080 INPUT "Chiave di ricerca :"
     ;a$

4090 IF LEN a$<1c THEN  LET a$=a
     $+" ": GO TO 4090
4100 FOR h=1 TO nr: IF m$(h,n)=a
     $ THEN  CLS : PRINT "RECORD
      N.";h'': FOR m=1 TO nc:
      PRINT c$(m);m$(h,m):
      NEXT m: IF p$<>"S"
      THEN  PRINT "PREMI UN TAST
     O": PAUSE 0: GO TO 4120
4110 IF m$(h,n)=a$ AND p$="S"
      THEN  LPRINT "RECORD N.";h
     '': FOR m=1 TO nc: LPRINT c
     $(m);m$(h,m): NEXT m:
      LPRINT '''
4120 NEXT h: PRINT "FINE LISTA":
      PRINT "PREMI UN TASTO":
      PAUSE 0: GO TO 120
```

DAL 1° DICEMBRE 1984

J. ADVERTISING E JACKSON PIU' VICINI!



Dal 1° dicembre 1984 J. ADVERTISING, concessionaria esclusiva di pubblicità per tutte le riviste del Gruppo Editoriale Jackson, Computer Publishing Group e J. Soft ha cambiato indirizzo. Il nuovo indirizzo è:

J. ADVERTISING s.r.l.
V.le Restelli, 5
20124 MILANO
Tel. 02/6882895-6882458-6880606

Sindisc 01

Sindisc 01a
Sindisc 01

di **Daniele Malavasi**

Una produzione per SUPERSINC DiscoVogue HARDWARE AND SOFTWARE DIFFUSION

Le caratteristiche

Sindisc è un gadget completamente automatizzato, che può suonare, nel modo da voi prescelto, fino a otto brani musicali. Lasciamo a voi la sorpresa di ascoltare e capire quali sono stati gli otto hits prescelti per questa prima compilation: diciamo solo che sono collocabili di buon diritto tra quelli più gettonati ed ascoltati dell'anno. Visto che la grafica, veramente notevole, e la musica, ben dosata, presentano molto eloquentemente le caratteristiche del programma, ci limitiamo a parlare delle modalità d'uso, che tra l'altro sono già inserite nel Sindisc stesso, in maniera molto curata.
All'inizio il programma mostra queste istruzioni, per passare poi, dopo qualche settaggio grafico, alla elencazione degli otto brani disponibili. Comparirà inoltre il monitor di controllo, composto da sei voci (nell'ordine UP, DOWN, PLAY, STOP, TRACK, REPEAT), che sono poi le altrettante opzioni disponibili: ad ognuna di queste è associato un numero (dall'uno al sei), che è relativo al tasto da premere sulla keyboard dello Spectrum per richiamare l'opzione.
La prima cosa che si fa notare è il cursore lampeggiante rosso-giallo, che indica il brano prescelto e che comparirà in alto a sinistra sopra al primo brano della scaletta musicale. Esso andrà spostato su e giù, premendo i tasti 1 e 2 dello Spectrum e cioè servendosi delle relative opzioni UP e DOWN. Il cursore può scorrere, a vostra discrezione, tutto il menu dei titoli disponibili: quando volete ascoltarne uno in particolare, fermate il cursore sul brano prescelto (per ognuno vedrete indicati titolo e relativo interprete) ed entrate quindi nella funzione PLAY premendo 3; il brano verrà eseguito dall'inizio alla fine, dopodiché Sindisc ritornerà automaticamente sullo STOP. Se volete che il brano sia ripetuto continuamente all'infinito, basta premere il REPEAT (tasto 6) prima di PLAY (tasto 3). Facile come usare un normale registratore!

È anche possibile poi ascoltare tutto il programma disponibile (e cioè gli otto brani, dal primo all'ultimo) scegliendo l'opzione TRACK tramite il tasto 5; vi garantirete in questo modo quasi dieci minuti di musica, che potrete anche in questo caso risentire un numero infinito di volte premendo il REPEAT (tasto 6) prima del TRACK (tasto 5).
Il monitor di controllo in tempo reale segnala con estrema chiarezza la situazione da voi creata, cosicché non c'è il rischio di "perdersi" in operazioni misteriose ed indefinite. Altre "facility", che si riveleranno di particolare importanza, sono:
a) l'opzione STOP (tasto 4), mediante la quale potete fermare un'esecuzione musicale in qualsiasi momento e passare così da una funzione all'altra con la massima rapidità;
b) la possibilità di rivedere, quando il programma è sullo STOP, le istruzioni d'uso semplicemente premendo il tasto P.
Tutto qui: non rimane altro da fare che passare alla realizzazione del supporto che, nonostante l'ampiezza dei relativi listati (circa 28 videate), non dovrebbe presentare alcuna difficoltà.

Battitura e salvataggio

Adesso che il modus operandi dovreb-

be essere a grandi linee abbastanza chiaro, entriamo nel cuore dell'argomento, illustrando i modi e le tecniche più opportune per salvare su cassetta i due listati. Sindisc è composto da due sottoprogrammi, che sono il sottoprogramma di lancio e il master. Questi dovranno essere sistemati consecutivamente l'uno dopo l'altro nella cassetta, come mostra la figura seguente.

Il lavoro va iniziato battendo innanzitutto il listato 1, che sarà salvato sulla cassetta con l'istruzione SAVE "SINDISC 01" LINE 1. Accertatisi (tramite VERIFY) della corretta memorizzazione su nastro, si procederà poi a battere il programma master e a salvarlo sulla stessa cassetta tramite l'istruzione SAVE "MASTER" LINE 1, usando l'accortezza di farlo seguire al sottoprogramma di lancio solo dopo tre-quattro secondi. Se il listato master vi sembra troppo lungo per le vostre capacità di copiatura o per altri motivi, potete ridurne l'ampiezza dalle originali 28 a circa 22 videate, rinunciando a battere le linee 524, 525, 1034 e dalla 6009 alla 6099, che sono quelle relative alle istruzioni. Il risparmio di tempo (e di battitura) sarà del 20% circa (sono 6 videate in meno), ma francamente questo metodo non sembra essere molto opportuno visto che il programma, essendo le istruzioni d'uso molto ben presentate e comunque sempre utili all'eventuale utilizzatore casuale, ne verrebbe mortificato.

Una volta verificato l'esatto caricamento dei listati si può procedere alla verifica pratica, caricando il programma con LOAD "" e verificando l'assenza di stranezze logiche e/o musicali.

Descrizione del programma

Linee	Commento
Listato 1: SOTTOPROGRAMMA DI LANCIO	
10	inizio del sottoprogramma; preparazione del video
11	istruzioni di carica del master
Listato 2: MASTER	
5-9	inizio del Sindisc, predisposizione grafica del video
522-602	preparazione grafica del campo d'azione
801	determinazione delle variabili e del cursore
1030-1099	routine principale di gestione
2510-2599	routine della funzione TRACK
3005-3099	routine della funzione PLAY
3510-3599	routine della funzione REPEAT
4010-4099	aggiornamento e stampa del cursore
5110-5199	data-set del primo brano
5210-5299	data-set del secondo brano
5310-5399	data-set del terzo brano
5410-5499	data-set del quarto brano
5510-5599	data-set del quinto brano
5610-5699	data-set del sesto brano
5710-5799	data-set del settimo brano
5810-5899	data-set dell'ottavo brano
6009-6099	istruzioni d'uso
8010-8099	predisposizione dei caratteri grafici e delle stringhe
9710-9799	data-set per la grafica del video
9832-9890	set dei microcaratteri

Dove usare Sindisc

Pur sapendo benissimo che la fantasia è una virtù che non ha bisogno di stimoli per essere esplicata, vi suggeriamo di non limitare il Sindisc ad un ascolto "personal", ma di sfruttarlo nelle innumerevoli occasioni che le feste di fine anno vi offriranno.

Il party di San Silvestro potrebbe essere l'occasione azzeccata: sistematelo a tal scopo in un angolino della sala, creando una specie di saletta con divani e poltrone: qualsiasi amico o invitato potrà usarlo senza difficoltà, anche se già reso allegro da spumante o champagne; infatti, premendo semplicemente il tasto P, è possibile rivedere in qualsiasi momento tutte le istruzioni d'uso (ammesso ovviamente che le abbiate inserite durante la battitura).

Certi che la vostra fantasia avrà già partorito idee migliori, preferiamo abbandonare il discorso, ricordando solo che la particolare struttura del programma rende il Sindisc adattabile ad eventuali futuri inserimenti di altre canzoni, motivo per cui non escludiamo la pubblicazione di una serie di articoli e di relative linee sostitutive atte a correggere il programma originale qui presentato e a rendere possibile la realizzazione di una serie di Sindisc adatta a formare una soft-discoteca veramente originale. Saranno molto gradite a tal riguardo vostre impressioni e richieste (si accetta tutto, dal liscio all'inno di Mameli).

Sindisc 01
versione per ZX Spectrum 48K
programma 1

```
10 PAPER 0: BORDER 0: CLS :
   PRINT #1;AT 0,0; INK 1;"
   {G5}{30G3}{SG5}{G1}{30G3}
   {G2}";AT 0,10; INK 6;" SIND
   ISC 01 ";AT 1,5; INK 7;" ©
   1984 by DISCOVOGUE "
11 FOR n=1 TO 3: BEEP .1,20:
   BEEP .1,23: BEEP .1,37:
   NEXT n: PAUSE 30: LOAD ""
```

Sindisc 01
versione per ZX Spectrum 48K
programma 2

```
5 PAPER 0: BORDER 0: CLS :
  RESTORE 8010: GO SUB 8000:
  FOR n=1 TO 13: READ z:
  PRINT AT z,1; PAPER 1;r$:
  NEXT n: POKE 23658,255
9 FOR b=1 TO 6: INK b:
  PAPER 0: PRINT AT 4,0;a$:
  FOR n=5 TO 17: PRINT AT n,
  0;"A";AT n,31;"A": BEEP .01
  ,n*b/1.5: NEXT n: PRINT
  AT 18,0;a$;AT 21,8; INK b+1
  ; FLASH 1; BRIGHT 1;f$:
  NEXT b
```

```
522 GO SUB 9700: PRINT AT 21,8;
    INK 0; PAPER 0;f$: FOR n=1
    TO 10: BEEP .002,50:
    NEXT n
523 PRINT AT 21,0; INK 1;c$(1
    TO 10);AT 21,22;c$(11 TO 2
    0);#1;AT 0,0; INK 1;c$(21
    TO 25);AT 0,27;c$(26 TO 30
    ): PRINT AT 21,10; INK 6;;"
    SINDISC 01 ";#1;AT 0,5;
    INK 7;" © 1984 by DISCOVOG
    UE "
524 GO TO 6060
525 LET xx=55: LET yy=18: FOR n
    =1 TO 38: GO SUB 9800+
    CODE i$(n): BEEP .003,55:
    LET xx=xx+4: NEXT n
530 DATA 13,109,13,85,13,61,13,
    37,133,109,133,85,133,61,13
    3,37,149,149,213,149:
    RESTORE 530: INK 4: FOR n=
    1 TO 8: READ x: READ y:
    PLOT x,y: DRAW 109,0:
    DRAW 0,21: DRAW -109,0:
    DRAW 0,-21: BEEP .1,(x/2+y
    )/3: NEXT n
531 READ x: READ y: PLOT x,y:
    DRAW 61,0: DRAW 0,21:
    DRAW -61,0: DRAW 0,-21:
    READ x: READ y: PLOT x,y:
    DRAW 42,0: DRAW 0,21:
    DRAW -42,0: DRAW 0,-21:
    BEEP .3,20
599 DATA 17,113,21,28,17,89,37,
    45,17,65,60,69,17,41,79,90,
    137,113,111,128,137,89,137,
    146,137,65,159,166,137,41,1
    86,193,17,121,1,20,17,97,29
    ,36,17,73,46,59,17,49,70,78
    ,137,121,91,110,137,97,129,
    136,137,73,147,158,137,49,1
    67,185,153,161,194,153,153,
    197,185,161,202,185,153,207
    ,217,161,212,217,153,220,16
    1,161,195,196,161,153,198,2
    01,193,161,203,206,193,153,
    208,211,225,161,213,219,225
    ,153,221,226
600 FOR v=1 TO 2: FOR n=1 TO 8:
    READ xx: READ yy: READ i:
    READ f: FOR g=i TO f:
    INK 3+v*2: GO SUB 9800+
    CODE m$(g): LET xx=xx+4::
    NEXT g: BEEP .05,v*20+n:
    NEXT n: NEXT v
601 FOR n=1 TO 6: READ xx:
    READ yy: READ i: INK 6:
    GO SUB 9800+CODE m$(i):
    LET xx=xx+4: BEEP .5,n*5:
    NEXT n
602 FOR n=1 TO 6: READ xx:
    READ yy: READ i: READ f:
    INK 3: FOR g=i TO f:
    GO SUB 9800+CODE m$(g):
    LET xx=xx+4: NEXT g:
    BEEP .05,n*2: NEXT n
801 LET cr=1: LET ccr=6: LET cc
    c=2: PRINT AT 2,24; OVER 1;
    INK 7; BRIGHT 1;"  ";AT 6,
    2; BRIGHT 0; FLASH 1;
    PAPER 2; INK 6;z$;AT 7,2;z
    $
1030 LET k=CODE INKEY$: IF k=49
     OR k=50 THEN  LET tr=0:
     GO SUB 4000
1031 IF k=51 THEN  LET tr=0:
     GO SUB 3000
1032 IF k=53 THEN  LET tr=1:
     GO SUB 2500
1033 IF k=54 THEN  GO SUB 3500
1034 IF k=80 THEN  BEEP .3,50:
     PAUSE 20: PRINT AT 19,1;
     PAPER 0;r$: FOR n=5 TO 17:
     PRINT AT n,1; PAPER 1;r$:
     NEXT n: GO TO 6060
1099 GO TO 1001
2510 PRINT AT ccr,ccc; OVER 1;
     PAPER 1; INK 7;z$;AT ccr+1
     ,ccc; INK 5;z$;AT 1,28;
     PAPER 0; INK 7; BRIGHT 1;"
     ": LET ccr=6: LET ccc=2
2580 FOR h=1 TO 8: LET cr=h:
     GO SUB 4000: PAUSE 35:
     GO SUB 3000: PAUSE 35:
     NEXT h
2585 IF rep=2 THEN  GO TO 2580
2590 PRINT AT 1,28; INK 3;
     BRIGHT 0; OVER 1;"  ";
     AT 1,24;"  ";AT 2,24; INK 7
     ; BRIGHT 1;"  ": RETURN
3005 PRINT AT 1,24; INK 7;
     BRIGHT 1; OVER 1;"  ";AT 1
     ,20; INK 3; BRIGHT 0;"  ";
     AT 2,20;"  ";AT 2,24;"  ":
     BEEP .1,55: PAUSE 1:
     PAUSE 35
3009 RESTORE 5010+cr*100: READ s
     a: READ v1: FOR p=1 TO sa*2
     STEP 2: READ pe: READ vo:
     LET m(p)=pe: LET m(p+1)=vo
     : NEXT p
```

```
3010 FOR r=1 TO v1: FOR p=1 TO s
     a*2 STEP 2: LET pe=m(p):
      LET vo=m(p+1): FOR v=1
      TO vo: RESTORE 5000+cr*100
     +pe: READ no: FOR n=1 TO no
     : READ du: READ fr: READ pa
     : BEEP du/100,fr
3020 LET o=CODE INKEY$: IF o=52
      THEN  GO TO 3099
3090 PAUSE pa: NEXT n: NEXT v:
      NEXT p: NEXT r: IF rep=1
      AND tr=0 THEN  GO TO 3099
3095 IF rep=2 AND tr=0 THEN
      GO TO 3010
3096 IF tr=1 THEN  RETURN
3099 PRINT AT 1,24; OVER 1;
      INK 3;"  ";AT 2,28;"  ";
     AT 1,28;"  ";AT 2,24;
      INK 7; BRIGHT 1;"  ":
      GO TO 1000
3510 LET rep=3-rep
3511 PRINT AT 2,28; OVER 1;
      INK 4*rep-1; BRIGHT rep-1;
     "  "
3599 BEEP .1,55: RETURN
4010 IF k=49 THEN  LET cr=cr-1:
      IF cr<1 THEN  LET cr=1
4011 IF k=50 THEN  LET cr=cr+1:
      IF cr>8 THEN  LET cr=8
4012 LET x=CODE d$(cr*2-1)-40:
      LET y=CODE d$(cr*2)-40
4013 IF ccr<>x AND tr=0 THEN
      PRINT AT k-48,20; INK 7;
      OVER 1; BRIGHT 1;"  "
4020 IF ccr<>x THEN  PRINT
     AT ccr,ccc; OVER 1; PAPER 1
     ; INK 7;z$;AT ccr+1,ccc;
      INK 5;z$
4021 PRINT AT x,y; OVER 1;
      FLASH 1; PAPER 2; INK 6;z$
     ;AT x+1,y;z$
4022 IF ccr<>x THEN  BEEP .01,
     30
4080 LET ccr=x: LET ccc=y
4090 IF ccr=x THEN  PRINT AT 1,2
     0; OVER 1; INK 3; BRIGHT 0;
     "  ";AT 2,20;"  "
4099 RETURN
5110 DATA 15,2,17,1,17,1,11,1,11
     ,1,12,1,13,1,13,1,12,1,13,1
     ,13,1,14,1,15,1,14,1,16,1,1
     7,1
5111 DATA 15,14,2,2,14,7,2,14,10
     ,2,20,9,4,20,7,4,14,2,2,14,
     0,2,14,0,2,14,-2,2,14,0,2,4
     0,2,16,40,10,2,56,9,12,40,5
     ,2,56,7,12
5112 DATA 7,14,-5,2,14,-3,2,14,-
     2,2,14,0,2,30,2,6,14,-5,2,4
     0,-5,8
5113 DATA 2,14,7,2,30,5,12
5114 DATA 13,30,7,6,14,7,2,20,7,
     4,30,7,6,20,7,4,14,7,2,30,7
     ,6,14,2,2,14,7,2,14,9,2,14,
     10,2,14,10,2,40,9,8
5115 DATA 2,30,5,6,40,7,8
5116 DATA 4,14,5,2,14,9,2,14,5,2
     ,56,7,12
5117 DATA 13,14,-5,2,14,-3,2,14,
     -2,2,14,0,2,30,2,6,14,3,2,5
     6,0,12,30,0,6,14,2,2,56,-2,
     12,30,-2,6,14,0,2,56,-5,12
5210 DATA 10,2,11,1,12,1,13,1,14
     ,1,13,1,15,1,16,1,17,1,18,1
     ,15,1
5211 DATA 16,14,19,2,14,17,2,14,
     19,2,20,22,4,20,19,4,20,17,
     4,40,17,8,20,15,4,20,17,4,2
     0,15,4,20,15,4,14,15,2,14,1
     7,2,14,15,2,10,14,2,40,12,8
5212 DATA 14,14,19,2,14,17,2,14,
     19,2,20,22,4,20,19,4,20,17,
     4,40,17,8,20,15,4,20,17,4,2
     0,15,4,20,15,4,14,15,2,10,1
     4,2,40,12,8
5213 DATA 10,20,20,4,14,22,2,14,
     24,2,14,22,2,20,19,4,20,17,
     4,40,15,8,12,19,4,12,17,4,2
     4,17,8
5214 DATA 3,14,17,2,14,15,2,28,1
     5,4
5215 DATA 4,40,24,8,14,22,2,14,2
     4,2,30,22,18
5216 DATA 9,20,24,4,40,22,8,40,2
     0,8,14,20,2,14,20,2,14,19,2
     ,14,20,2,14,22,4,28,24,4
5217 DATA 10,20,24,4,40,22,8,40,
     20,8,14,20,2,14,20,2,14,20,
     2,20,20,4,20,19,4,20,17,4,4
     0,15,8
```

```
5218 DATA 11,20,24,4,40,22,8,40,
     20,8,14,20,2,14,20,2,14,19,
     2,20,20,4,20,17,4,14,19,2,1
     4,17,2,28,17,4
5310 DATA 13,2,11,1,12,1,11,1,13
     ,1,14,1,15,1,14,1,16,1,17,1
     ,18,1,19,1,18,1,20,2
5311 DATA 9,10,5,2,10,5,2,10,4,2
     ,10,5,2,20,4,4,20,2,4,14,-2
     ,2,14,2,2,28,4,4
5312 DATA 9,10,9,2,10,9,2,10,7,2
     ,10,9,2,20,5,4,20,7,4,14,5,
     2,14,7,2,28,9,4
5313 DATA 7,10,9,2,10,9,2,10,7,2
     ,10,9,2,14,5,2,14,7,2,28,9,
     4
5314 DATA 14,10,12,2,10,12,2,10,
     5,2,10,12,2,20,5,4,20,10,4,
     14,5,2,14,2,2,14,10,2,10,9,
     2,10,9,2,10,7,2,10,9,2,20,5
     ,4
5315 DATA 4,20,7,4,14,4,2,14,7,2
     ,28,12,4
5316 DATA 2,20,7,2,40,9,8
5317 DATA 3,14,0,12,14,0,2,28,2,
     4
5318 DATA 20,14,2,2,14,2,2,14,0,
     2,14,4,2,14,4,2,14,2,2,20,0
     ,4,40,2,8,14,2,2,14,9,2,14,
     7,2,10,5,2,10,2,2,10,7,2,20
     ,5,4,14,4,2,14,4,2,28,2,4,2
     0,4,4,20,4,4
5319 DATA 1,40,5,8
5320 DATA 13,40,9,8,14,12,2,14,5
     ,2,28,5,4,10,5,2,10,9,2,10,
     7,2,10,5,2,20,7,4,40,9,8,14
     ,12,2,14,5,2,28,5,4
5410 DATA 4,3,11,1,12,1,13,1,14,
     1
5411 DATA 14,15,19,2,15,17,2,15,
     15,2,25,17,9,10,19,7,25,19,
     9,15,19,2,15,22,2,15,19,2,2
     5,17,9,15,15,2,25,17,9,15,1
     5,2,25,12,9
5412 DATA 15,15,12,2,15,19,2,15,
     19,2,15,15,2,25,17,9,15,17,
     2,25,17,9,15,22,2,15,22,2,1
     5,19,2,25,17,9,15,15,2,25,1
     7,9,15,15,2,25,12,9
5413 DATA 14,15,10,2,15,22,2,15,
     20,2,30,19,4,15,20,2,25,15,
     9,15,19,2,30,17,4,15,20,2,1
     5,19,2,30,17,4,30,15,4,15,1
     7,2,15,19,19
5414 DATA 11,15,22,2,15,20,2,25,
     19,9,15,20,2,25,15,9,15,19,
     2,25,17,9,15,20,2,15,19,2,3
     0,17,4,25,15,26
5510 DATA 8,2,11,1,12,1,13,1,14,
     1,14,1,15,1,14,1,15,1
5511 DATA 8,20,13,4,30,16,6,20,1
     6,4,20,16,4,20,16,4,20,16,4
     ,20,14,4,40,13,8
5512 DATA 6,20,13,4,30,16,6,20,1
     6,4,20,16,4,20,18,4,40,16,8
5513 DATA 14,20,16,4,20,16,4,20,
     14,4,20,14,4,20,13,4,20,13,
     4,20,11,4,30,9,6,20,11,4,20
     ,13,4,20,14,4,20,9,4,20,11,
     4,40,13,8
5514 DATA 7,5,9,1,5,9,1,5,9,1,5,
     9,1,5,9,1,20,11,4,40,13,8
5515 DATA 15,5,9,1,5,9,1,5,9,1,5
     ,9,1,5,9,1,20,11,4,20,13,4,
     20,6,4,20,9,4,20,11,4,20,13
     ,4,20,6,4,20,9,4,20,11,4,40
     ,13,16
5610 DATA 7,2,11,1,12,1,13,1,11,
     1,12,1,14,1,15,1
5611 DATA 18,14,9,2,14,9,2,14,10
     ,2,14,9,2,40,12,8,14,9,2,14
     ,10,2,14,9,2,40,12,8,14,9,2
     ,14,7,2,14,7,2,14,5,2,14,8,
     2,20,7,4,14,7,2,14,5,2,40,7
     ,8
5612 DATA 16,14,0,2,20,7,4,20,7,
     4,14,7,2,14,9,2,14,7,2,40,1
     0,8,14,9,2,14,7,2,14,7,2,14
     ,5,2,14,8,2,14,7,2,14,7,2,1
     4,5,2,40,5,8
5613 DATA 4,20,3,4,20,3,4,14,3,2
     ,40,2,8
5614 DATA 9,14,9,2,20,5,4,14,0,2
     ,40,2,8,12,4,1,12,4,1,12,4,
     1,12,5,1,40,4,8
5615 DATA 9,14,9,2,20,5,4,14,0,2
     ,40,-2,8,12,7,1,12,7,1,12,7
     ,1,12,7,1,40,7,8
5710 DATA 12,2,11,1,12,1,12,1,13
     ,1,14,1,11,1,12,1,12,1,13,1
     ,15,1,16,1,14,1
5711 DATA 5,20,9,4,14,9,2,14,9,2
     ,20,5,4,20,7,4
5712 DATA 9,20,7,4,14,7,2,14,7,2
     ,20,10,4,20,9,4,14,9,2,14,9
     ,2,20,5,4,20,7,4
5713 DATA 7,14,7,2,14,7,2,14,7,2
     ,20,9,4,20,5,4,14,4,2,40,2,
     8
```

```
5714 DATA 5,20,9,4,20,7,4,16,5,4
     ,16,4,4,40,2,8
5715 DATA 11,14,0,2,14,0,2,14,4,
     2,14,4,2,14,4,2,20,4,4,14,2
     ,2,14,0,2,14,2,2,20,2,4,20,
     2,4
5716 DATA 9,14,0,2,14,0,2,14,4,2
     ,14,4,2,14,4,2,20,4,4,14,5,
     2,14,4,2,20,2,4

5810 DATA 18,2,11,1,12,1,13,1,12
     ,1,14,1,14,1,11,1,12,1,13,1
     ,12,1,14,1,15,1,15,1,16,1,1
     7,1,15,1,15,1,16,1
5811 DATA 8,30,-4,6,20,-6,4,30,-
     8,3,30,-6,6,20,-8,4,30,-6,4
     ,20,-1,4,30,-1,6

5812 DATA 7,20,-6,4,16,-4,2,16,-
     6,2,20,-8,4,20,-6,4,16,-8,2
     ,30,-8,6
5813 DATA 9,20,-4,4,16,-6,2,16,-
     8,2,20,-6,4,16,-8,2,16,-6,2
     ,16,-6,2,16,-1,2,30,-1,6
5814 DATA 18,16,-4,2,16,-4,2,16,
     -3,2,16,-3,2,16,-1,2,16,-1,
     2,20,4,4,20,-1,4,20,-1,4,16
     ,-1,2,30,-3,6,16,-4,2,16,-6
     ,2,16,-6,2,16,-6,2,16,-6,2,
     16,-8,2,30,-6,10
5815 DATA 5,16,-6,2,16,-6,2,16,-
     8,2,30,-8,6,30,-8,6
5816 DATA 8,16,-6,2,16,-6,2,16,-
     8,2,16,-8,2,16,-3,2,16,-3,2
     ,16,-4,2,30,-8,6
5817 DATA 12,16,-4,2,16,-4,2,16,
     -3,2,16,-3,2,16,-1,2,16,-1,
     2,16,4,2,16,4,2,16,4,2,16,8
     ,2,16,4,2,30,1,6
6009 LET i$="LE ISTRUZIONI":
      RETURN
6010 LET i$="DISCOVOGUE PRESENTA
      SINDISC 01, COMPILATION M
     USICALE PER SINCLAIR SPECTR
     UM.QUESTA RACCOLTA DISPONE
     DI OTTO HITS CHEPOTRETE ASC
     OLTARE QUANDO E COME":
      RETURN
6011 LET i$=" VOLETE CON L'AUSIL
     IO DEIPRATICISSIMI COMANDI
     ELETTRONICI, CHE SONO:1 UP2
      DOWN3 PLAY4 STOP5 TRACK6 R
     EPEATIL NUMERO INDICA IL TA
     STO DA PREMERE PER ENTRARE
     NELLACORRISPONDENTE FUNZION
     E.": RETURN
6012 LET i$="PER CONTINUARE PREM
     ERE UN TASTO": RETURN
6020 LET i$="OPZIONI DISPONIBILI
     :ASCOLTO DI UN PARTICOLARE
     BRANO:SELEZIONATELO TRAMITE
      LE FUNZIONI UP E DOWN:UN E
     VIDENZIATORELAMPEGGIANTE SE
     GNALERA' LA VOSTRA POSIZION
     E.PREMETE QUINDIPLAY.IL COR
     RISPONDENTE": RETURN

6021 LET i$="BRANO VERRA' SUBITO
      SUONATO.PREMENDOREPEAT PRI
     MA DI PLAY,IL BRANO SARA' R
     IPETUTO ALL'INFINITO.":
      RETURN

6022 LET i$="ASCOLTO DI TUTTI I
     BRANI:AVVIENE AUTOMATICAMEN
     TE SELEZIONANDO LA FUNZIONE
      TRACK:VER-RANNO ESEGUITI T
     UTTI I BRANI,DAL PRIMO ALL'
     ULTIMO.PREMENDOREPEAT PRIMA
      DI PLAY LA SEQUENZA VERRA'
      ESEGUITA ALL'INFINITO.":
      RETURN

6023 LET i$="PER CONTINUARE PREM
     ERE UN TASTO": RETURN

6024 LET i$="UN MONITOR DI CONTR
     OLLO SEGNALA IN TEMPO REALE
      TUTTE LE FUNZIONI PRESCELT
     E.TRAMITE LO STOP POTETE IN
      UN QUALSIASI MOMENTO INTER
     ROMPERE UNA ESECUZIONE MUSI
     CALE E PASSARE AD UN'ALTRA
     FUNZIONE,OPPURE,PREMENDO AN
     CHE P,RIVEDERE LE ISTRUZION
     I.": RETURN
6025 LET i$="PER CONTINUARE PREM
     ERE UN TASTO": RETURN

6050 DATA 1,20,130,1,14,3,10,122
     ,1,58,10,116,59,116,10,110,
     117,147,16,139,110,1,25,10,
     104,26,68,58,90,69,69,58,82
     ,73,73,114,90,79,79,114,82,
     85,85,170,90,91,91,170,82,9
     8,98
```

```
6051 DATA 66,90,71,72,66,82,75,7
     8,122,90,81,84,122,82,87,90
     ,178,90,93,97,178,82,100,10
     5,10,66,106,159,10,60,160,1
     83,1,66,18,1,31
6052 DATA 5,10,114,1,20,10,98,21
     ,52,10,90,53,112,10,84,113,
     170,10,78,171,192,2,101,78,
     1,36,10,72,37,94
6053 DATA 4,10,58,1,25,10,50,26,
     84,10,44,85,142,10,38,143,2
     04,1,66,18,1,31
6054 DATA 5,10,114,1,55,10,108,5
     7,75,10,95,76,134,10,89,136
     ,190,10,83,191,237,1,66,18,
     1,31
6060 INK 5: RESTORE 6050: FOR n=
     0 TO 3: READ r: GO SUB 6009
     +n: GO SUB 6099: NEXT n
6069 LET s$="3=2/3/26362=3="
6070 FOR n=3 TO 13 STEP 2:
      PRINT AT CODE s$(n)-40,
     CODE s$(n+1)-40; PAPER 1;
      INK 7; BRIGHT 1; OVER 1;"
      ";AT CODE s$(n-2)-40,
     CODE s$(n-1)-40; BRIGHT 0;
      INK 5;"     ";AT 19,8;
      INK CODE s$(n)-47;
      BRIGHT 1; PAPER 0;"
                        "
6071 IF INKEY$<>"" THEN
      PRINT AT CODE s$(n)-40,
     CODE s$(n+1)-40; OVER 1;
      PAPER 1;"     ";AT 19,1;
      OVER 0; PAPER 0;r$: BEEP
     3,50: PAUSE 20: FOR t=6
      TO 17: PRINT AT t,1;
      PAPER 1;r$: NEXT t:
      GO TO 6073
6072 BEEP .07,n: NEXT n: GO TO 6
     070
6073 FOR n=0 TO 3: READ r:
      GO SUB 6020+n: GO SUB 6099
     : NEXT n: LET gg=3
6074 LET gg=7-gg: PRINT AT 19,8;
      INK gg; BRIGHT 1; OVER 1;"
                        "
6075 IF INKEY$<>"" THEN
      PRINT AT 19,1; PAPER 0;r$:
      BEEP .3,50: PAUSE 20:
      FOR t=6 TO 17: PRINT AT t,
     1; PAPER 1;r$: NEXT t:
      GO TO 6089
6076 BEEP .07,30: GO TO 6074
6089 FOR n=4 TO 5: READ r:
      GO SUB 6020+n: GO SUB 6099
     : NEXT n
6090 LET gg=7-gg: PRINT AT 19,8;
      INK gg; BRIGHT 1; OVER 1;"
                        "
6091 IF INKEY$<>"" THEN
      PRINT AT 19,1; PAPER 0;r$:
      BEEP .3,50: PAUSE 20:
      FOR t=5 TO 17: PRINT AT t,
     1; PAPER 1;r$: NEXT t:
      LET i$="PER RIVEDERE LE IS
     TRUZIONI PREMERE ""P""":
      GO TO 525
6092 BEEP .07,30: GO TO 6090
6099 FOR l=1 TO r: READ xx:
      READ yy: READ i: READ f:
      FOR b=i TO f: GO SUB 9800+
     CODE i$(b): LET xx=xx+4:
      BEEP .003,yy/2: NEXT b:
      NEXT l: RETURN
8010 DATA 0,48,120,252,252,120,4
     8,0,5,6,7,8,9,10,11,12,13,1
     4,15,16,17: FOR n=USR "a"
      TO USR "a"+7: READ z:
      POKE n,z: NEXT n
8011 LET r$="
                        ": LET a$="AAAAA
     AAAAAAAAAAAAAAAAAAAAAAAAAAA
     ": LET c$="{G5}{18G3}{SG5}
     {G1}{8G3}{G2}"
8020 LET f$="FERMA  IL NASTRO":
      LET z$="                ":
      LET d$=".*1*4*7*.9194979":
      DIM m(36)
8099 LET m$="DON'T WANNA LOSE YO
     UELEPHANTMOVIN ONNOVECENTOD
     REAM          P. LION    THE
     NIGHTVALERIE DORECI VORREBB
     E UN AMICOANTONELLO VENDITT
     ISAD SONGELTON JOHNLA COLEG
     IALAGARY LOWSTATE OF THE NA
     TIONINDUSTRY1UP2DOWN3PLAY4S
     TOP5TRACK  6REPEAT":
      RETURN
9710 DATA 0,152,60,152,93,152,12
     0,152,0,157,93,157,0,162,33
     ,162,60,162,93,162,120,162:
      INK 2: RESTORE 9710: FOR n
     =1 TO 11: READ x: READ y:
      FOR b=y TO y+2: PLOT x,b:
      DRAW 23,0: NEXT b: NEXT n
```

```
9711 DATA 27,152,33,152,54,152,6
     0,152,81,152,87,152,120,152
     : RESTORE 9711: FOR n=1
      TO 7: READ x: READ y:
      FOR b=x TO x+2: PLOT b,y:
      DRAW 0,12: NEXT b: NEXT n
9712 DATA 0,160,21,155,27,167,81
     ,165,81,167,87,167,93,160,1
     14,155: RESTORE 9712: FOR n
     =1 TO 8: READ x: READ y:
      FOR b=x TO x+2: PLOT b,y:
      DRAW 0,2: NEXT b: NEXT n
9799 LET rep=1: INK 0: RETURN
9832 RETURN
9839 PLOT xx+1,yy+4: DRAW -1,-1:
      RETURN
9840 PLOT xx+1,yy: DRAW -1,1:
      DRAW 0,2: DRAW 1,1:
      RETURN
9841 PLOT xx+1,yy: DRAW 1,1:
      DRAW 0,2: DRAW -1,1:
      RETURN
9844 LET xx=xx+1: PLOT xx+1,yy+1
     : DRAW -1,-1: LET xx=xx+2:
      RETURN
9845 PLOT xx,yy+2: DRAW 2,0:
      RETURN
9846 PLOT xx+1,yy: LET xx=xx+3:
      RETURN
9849 PLOT xx,yy: DRAW 2,0:
      DRAW -1,0: DRAW 0,4:
      DRAW -1,-1: RETURN
9850 PLOT xx+2,yy: DRAW -2,0:
      DRAW 0,1: DRAW 2,2: DRAW 0
     ,1: DRAW -1,1: DRAW -1,-1:
      RETURN
9851 PLOT xx,yy: DRAW 1,0:
      DRAW 1,1: DRAW -1,1:
      DRAW 1,1: DRAW -1,1:
      DRAW -1,0: RETURN
9852 PLOT xx+2,yy: DRAW 0,4:
      DRAW -2,-2: DRAW 0,-1:
      DRAW 1,0: RETURN
9853 PLOT xx,yy: DRAW 1,0:
      DRAW 1,1: DRAW -1,1:
      DRAW -1,0: DRAW 0,2:
      DRAW 2,0: RETURN
9854 PLOT xx+1,yy+2: DRAW 1,-1:
      DRAW -1,-1: DRAW -1,1:
      DRAW 0,2:: DRAW 1,1:
      DRAW 1,0: RETURN
9856 PLOT xx,yy+1: DRAW 1,-1:
      DRAW 1,1: DRAW -2,2::
      DRAW 1,1: DRAW 1,-1:
      RETURN
9858 PLOT xx+1,yy: PLOT xx+1,yy+
     2: RETURN
9865 PLOT xx,yy: DRAW 0,3:
      DRAW 1,1: DRAW 1,-1:
      DRAW -1,-1: DRAW 1,0:
      DRAW 0,-2: RETURN
9866 PLOT xx,yy: DRAW 0,4:
      DRAW 1,0: DRAW 1,-1:
      DRAW -1,-1: DRAW 1,-1:
      DRAW -1,-1: RETURN
9867 PLOT xx+2,yy: DRAW -1,0:
      DRAW -1,1: DRAW 0,2:
      DRAW 1,1: DRAW 1,0:
      RETURN
9868 PLOT xx,yy: DRAW 0,4:
      DRAW 1,0: DRAW 1,-1:
      DRAW 0,-2: DRAW -1,-1:
      RETURN
9869 PLOT xx+2,yy: DRAW -2,0:
      DRAW 0,2: DRAW 2,0: DRAW -
     2,0: DRAW 0,2: DRAW 2,0:
      RETURN
9870 PLOT xx,yy: DRAW 0,2:
      DRAW 2,0: DRAW -2,0:
      DRAW 0,2: DRAW 2,0:
      RETURN
9871 PLOT xx+2,yy+2: DRAW 0,-2:
      DRAW -1,0: DRAW -1,1:
      DRAW 0,2: DRAW 1,1: DRAW 1
     ,0: RETURN
9872 PLOT xx,yy: DRAW 0,4:
      DRAW 0,-2: DRAW 2,0:
      DRAW 0,2: DRAW 0,-4:
      RETURN
9873 PLOT xx,yy: DRAW 0,4: LET x
     x=xx-2: RETURN
9874 PLOT xx,yy+1: DRAW 1,-1:
      DRAW 1,1: DRAW 0,3:
      RETURN
9875 PLOT xx,yy: DRAW 0,4:
      PLOT xx+1,yy+2: DRAW 2,-2:
      DRAW -2,2: DRAW 2,2: LET x
     x=xx+1: RETURN
9876 PLOT xx+2,yy: DRAW -2,0:
      DRAW 0,4: RETURN
```

```
9877 PLOT xx,yy: DRAW 0,3:
      DRAW 1,1: DRAW 1,-1:
      DRAW 1,1: DRAW 1,-1:
      DRAW 0,-3: LET xx=xx+2:
      RETURN
9878 PLOT xx,yy: DRAW 0,4:
      DRAW 1,0: DRAW 1,-1:
      DRAW 0,-3: RETURN
9879 PLOT xx,yy+1: DRAW 0,2:
      DRAW 1,1: DRAW 1,-1:
      DRAW 0,-2: DRAW -1,-1:
      RETURN
9880 PLOT xx,yy: DRAW 0,4:
      DRAW 1,0: DRAW 1,-1:
      DRAW -1,-1: RETURN
9881 PLOT xx+3,yy-1: DRAW -1,1:
      DRAW -1,0: DRAW -1,1:
      DRAW 0,2: DRAW 1,1: DRAW 1
     ,-1: DRAW 0,-2: LET xx=xx+1
     : RETURN
9882 PLOT xx,yy: DRAW 0,4:
      DRAW 1,0: DRAW 1,-1:
      DRAW -1,-1: DRAW 1,-1:
      DRAW 0,-1: RETURN
9883 PLOT xx,yy: DRAW 1,0:
      DRAW 1,1: DRAW -2,2:
      DRAW 1,1: DRAW 1,0:
      RETURN
9884 PLOT xx,yy+4: DRAW 1,0:
      DRAW 0,-4: DRAW 0,4:
      DRAW 1,0: RETURN
9885 PLOT xx,yy+4: DRAW 0,-4:
      DRAW 2,0: DRAW 0,4:
      RETURN
```

```
9886 PLOT xx,yy+4: DRAW 0,-3:
      DRAW 1,-1: DRAW 1,1:
      DRAW 0,3: RETURN
9887 PLOT xx,yy+4: DRAW 0,-3:
      DRAW 1,-1: DRAW 1,1:
      DRAW 0,1: DRAW 0,-1:
      DRAW 1,-1: DRAW 1,1:
      DRAW 0,3: LET xx=xx+2:
      RETURN
9888 PLOT xx,yy: DRAW 4,4:
      DRAW -2,-2: DRAW -2,2:
      DRAW 4,-4: LET xx=xx+2:
      RETURN
9889 PLOT xx+1,yy: DRAW 0,2:
      DRAW -1,0: DRAW 0,2:
      DRAW 0,-2: DRAW 2,0:
      DRAW 0,2: RETURN
9890 PLOT xx,yy+4: DRAW 2,0:
      DRAW 0,-1: DRAW -2,-2:
      DRAW 0,-1: DRAW 2,0:
      RETURN
```

ERRATA CORRIGE

A causa di un disguido, nell'articolo "Come realizzare un'interfaccia", pubblicato nel numero 5 di SuperSINC, sono state tralasciate le didascalie delle figure 1 e 5, contenenti la lista dei componenti utilizzati. Ecco le parti mancanti delle didascalie.

Fig. 1
Trasformatore 220-6V 1A (non visibile in figura)
p = ponte raddrizzatore da 1A
X1 = regolatore 7805
C1 = condensatore elettrolitico 1000 μF 25V
C2 = condensatore elettrolitico 22 μF 12V
C3 = condensatore elettrolitivo 0.1 μF

Fig. 5
L = LED rossi
R = resistenze da 1200 Ω
C = condensatore 0.1 μF
I due circuiti integrati presenti in figura sono 7400

Ci scusiamo dell'inconveniente con tutti i lettori.

OGNI SETTIMANA IN EDICOLA 2 DISPENSE

FA SCUOLA

Enciclopedia di Elettronica e Informatica

✓ I temi affascinanti della civiltà del computer, gli sviluppi della società tecnologica in un opera creata per capire e affrontare il micromillennio.

✓ **L'enciclopedia giovane e pratica,** che nasce dai progressi della ricerca, che parla il linguaggio chiaro e conciso della "bit generation".

✓ **Lo strumento base per chi studia,** per chi lavora, per chi vuol vivere da protagonista le affascinanti novità del nostro tempo e prepararsi a quelle del futuro prossimo venturo.

In edicola
60 dispense
30 appuntamenti settimanali con gli esperti JACKSON e i tecnici TEXAS INSTRUMENTS

✓ **Ogni settimana:**
56 pagine di elettronica, informatica e comunicazioni.

✓ **In sole 30 settimane** una splendida opera per la vostra **biblioteca:**
7 prestigiosi volumi
1600 pagine complessive
700 foto e 2200 disegni a colori

GRUPPO EDITORIALE JACKSON

In collaborazione con il Learning Center
TEXAS INSTRUMENTS

LE PRIME 2 DISPENSE SOLO LIRE 2.500

TUTTO SU...

Un'opera prestigiosa

Elettronica – Informatica – Comunicazioni

- l'unica cosi completa e organica
- l'unica fatta da veri specialisti
- l'unica che non sarà mai sorpassata dai fatti

nuovidea

GRUPPO EDITORIALE JACKSON

In collaborazione con il Learning Center
TEXAS INSTRUMENTS

Flatman

di **J. Dave Rogers**
trad. e adatt.
di **Angelo Motta**

La vita rischiosa e movimentata di un... pulitore di finestre assunto dallo ZX81

Siete incaricati di tenere in ordine le finestre di un palazzo. Il lavoro consiste nel pulire le finestre sporche (rappresentate da un carattere grafico) (e riparare quelle rotte contrassegnate dalle 'X'), usando i secchi d'acqua e le lastre di vetro situate attorno al palazzo. Le difficoltà, come sempre, sono molteplici: non è possibile trasportare sia l'acqua che il vetro contemporaneamente, e le finestre, per agenti atmosferici, vibrazioni e vandalismo, si sporcano o si rompono in continuazione. Se riuscirete a pulire e riparare tutte le finestre perfettamente, passerete ad un livello di gioco più impegnativo.

Istruzioni per il gioco

Si usino i tasti cursore (5-6-7-8) per muoversi nelle direzioni riportate dagli stessi. Gli unici passaggi consentiti sono le scale, rappresentate dagli '=' in campo inverso; in caso contrario cadrete dal palazzo, con conseguente perdita di una vita delle quattro a disposizione in totale.
Per prendere i secchi d'acqua muoversi verso i simboli "W"; per le lastre di vetro verso i simboli "/".
Si può prendere quanta acqua o quanto vetro si vuole, ma non ambedue contemporaneamente: troppo pericoloso! Per cui, se avete l'acqua e volete prendere il vetro, dovrete lasciare i secchi d'acqua, che andranno persi.
Per pulire od aggiustare le finestre basta muoversi verso di esse. Con un secchio d'acqua o con una lastra di vetro potrete sistemare 4 finestre. L'uomo lampeggerà, evidenziando ciò che state trasportando o, se non avete niente, a vuoto.

Descrizione del programma

Linee	Commento
20-70	subroutine relativa al movimento dell'uomo. È la parte del programma più usata, le cui variabili sono state definite all'inizio del programma, e pertanto, vi è una notevole velocità di esecuzione della stessa, che non mancherà di stupire
40	controlla se l'uomo può muoversi, se prende acqua o vetro, o se sta riparando una finestra. A tale scopo viene controllata la locazione dello schermo scelta dalla mossa ed in base al carattere in essa contenuta viene determinata l'azione da seguire tramite la routine contenuta nelle linee da 300 a 580. Tali linee contengono anche dei RETURN per la chiamata della subroutine originaria di mossa del pulitore
100-110	creano un ciclo che costantemente chiama la subroutine di mossa. Questo ciclo viene occasionalmente interrotto e, tramite le linee 200-290, le finestre vengono sporcate o rotte in continuazione
310	usa l'algebra Booleana e consente di evitare diverse istruzioni IF-THEN-GOTO
600-690	inizializzano le variabili
700-892	formano la subroutine di stampa del palazzo a video per ciascun livello di gioco
705	stabilisce un numero casuale diverso ad ogni livello di gioco successivo
900-997	routine di caduta del pulitore e, in caso di ultimo a disposizione, richiesta di successiva gara

Modifiche al programma

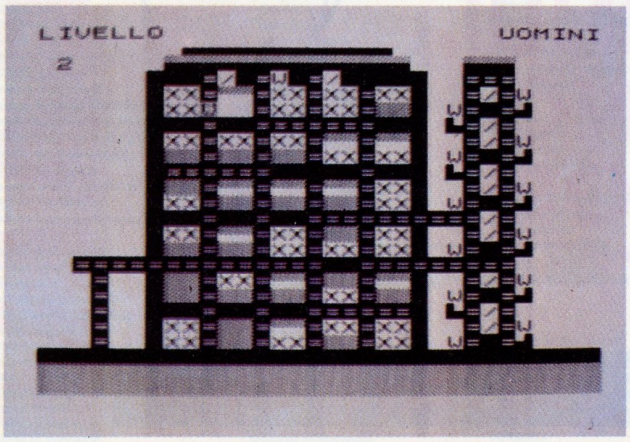
Nel programma sono state omesse le istruzioni sopra riportate per non allungarne la stesura, operazione comunque semplicissima.
Se il gioco sembra troppo difficile, si può variare la linea 800 inserendo 32 caratteri '=' in campo inverso al posto degli spazi inversi, in modo da permettere all'uomo ulteriori accessi al palazzo. Se il gioco sembra invece troppo facile, è possibile decrementare il valore 0,94 alla linea 110, in modo da aumentare la frequenza di rottura delle finestre.

Descrizione delle variabili

PP	posizione attuale dell'uomo
S	prossima posizione a video dell'uomo
X	inizio Display File
K	carattere dell'uomo
Y	carattere contenuto nella locazione a video scelta per la successiva mossa
W	contatore dei secchi d'acqua
GL	contatore delle lastre di vetro
PW	contatore delle finestre pulite
LE	livello di gioco
E,F,H	utilizzate per generare casualmente il tipo di incoveniente e la finestra interessata

Flatman versione per ZX81 16K

```
   1 REM FLATMAN
   8 GOTO 600
  10 REM
  11 REM [illegible]
  12 REM
  20 POKE PP,148
  30 LET S=PP+(INKEY$="8")-(INKE
Y$="5")+(33 AND INKEY$="6")-(33
AND INKEY$="7")
  40 IF PEEK S<>148 THEN GOTO 30
0
  50 LET PP=S
  60 POKE S,K
  70 RETURN
 100 GOSUB 20
 110 IF RND<.94 THEN GOTO 100
 200 LET E=5+INT (RND*6)*3
 210 GOSUB 20
 220 LET F=7+INT (RND*5)*3
 230 GOSUB 20
 240 IF RND>.8 THEN LET H=8+(53
AND H=8)
 250 GOSUB 20
 260 IF PEEK (X+E*33+F+1)=90 THE
N LET PW=PW-1
 270 GOSUB 20
 280 PRINT AT E,F;CHR$ H+CHR$ H
 290 GOTO 100
 297 REM
 298 REM SUBROUTINE
```



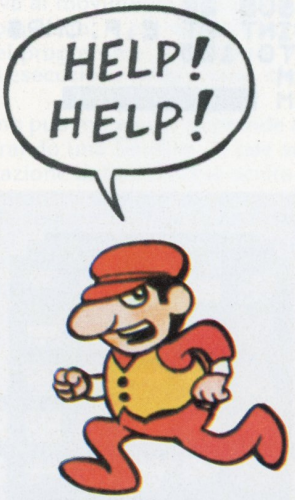

```
 299 REM
 300 LET Y=PEEK S
 310 GOTO 320+(80 AND Y=60)+(110
 AND Y=24)+(180 AND Y=8 AND W>0)
+(200 AND Y=61 AND GL>0)
 320 IF Y=0 OR Y=128 AND S-X<660
 THEN GOSUB 900
 330 RETURN

 400 LET W=W+4
 410 LET GL=0
 420 GOTO 450
 430 LET GL=GL+4
 440 LET W=0
 450 POKE S,90
 460 LET K=Y
 470 RETURN

 500 LET W=W-1
 510 GOTO 530
 520 LET GL=GL-1
 530 IF W=0 AND GL=0 THEN LET K=
0
 540 POKE S,90
 550 POKE (S+1-(2 AND PEEK (S-1)
=Y)),90
 560 LET PW=PW+1
 570 IF PW=30 THEN GOSUB 700
 580 RETURN

 597 REM
 598 REM INITIALIZATIONS
 599 REM
 600 LET X=PEEK 16396+256*PEEK 1
6397
 610 LET S=X+498
 620 LET PP=S
 630 LET K=23

 640 LET Y=1
 650 LET W=0
 660 LET GL=0
 670 LET H=8
 680 LET LE=0
 681 POKE 16418,0
 682 LET G$="[illegible]"

 683 LET LIFE=4
 684 LET L$="  ***"
 685 GOSUB 700
 690 GOTO 100
 697 REM

 698 REM PREPARATIONS [illegible]
 699 REM
 700 LET LE=LE+1
 705 RAND LE*333
```

```
710 LET PW=0
720 PRINT AT 1,8;"                    "
730 PRINT TAB 7;"                    
    "
740 PRINT TAB 6;"    /  W  /    
    "
750 FOR N=1 TO 6
760 PRINT TAB 6;"                    
    /  W"
770 PRINT TAB 6;"XXXXXXXXXXXXXX
  W    "
780 PRINT TAB 6;"                    
    /  "
790 NEXT N
800 PRINT AT 21,0;"                    
                    ";G$;G$
810 LET P$="      "
820 PRINT AT 6,13;P$;AT 9,7;P$;
AT 18,15;P$;AT 12,15;P$+P$( TO 4
);AT 15,3;P$+P$
830 IF LE<7 THEN PRINT AT 21-LE
*3,10;P$;"      ";P$
840 FOR N=0 TO 20
850 PRINT AT 0,N;"  W   W"
860 PRINT AT 5+INT (RND*6)*3,7+
INT (RND*5)*3;CHR$ 8+CHR$ 8
870 PRINT AT 0,N;"  M   M"
880 IF N>14+LE THEN PRINT AT N,
2;"W W"
890 NEXT N
891 PRINT AT 0,0;"LIVELLO
                UOMINI";AT 2,1;LE
892 GOTO 970
897 REM
898 REM ROUTINE DELLA TACCITA
899 REM

900 POKE S,Y
910 LET S=S+33
920 LET Y=PEEK S
930 POKE S,23
940 IF S-X<725 THEN GOTO 900
950 LET PP=X+499
960 LET LIFE=LIFE-1
970 PRINT AT 2,28;L$( TO LIFE)+
 "
980 IF LIFE>0 THEN RETURN
990 PRINT AT 3,7;"GAME OVER"
993 PRINT AT 23,0;"  PREMI UN'L
INE PER ALTRA GARA  "
995 IF INKEY$<>CHR$ 118 THEN GO
TO 995
997 GOTO 600
1000 SAVE "FLATMAN"
1010 PRINT AT 3,10;"FLATMAN",,,"
FERMA IL REGISTRATORE E PREMI
NEWLINE",,,"PER MUOVERE USA I TA
STI CON LE  FRECCE (5-6-7-8)"
1020 PRINT AT 3,7;"
 "
1030 IF INKEY$<>CHR$ 118 THEN GO
TO 1010
1040 CLS
1050 RUN
```


IN MILANO - VIA MASCHERONI, 14

IL VOSTRO
"NUOVISSIMO"
COMPUTER SHOP

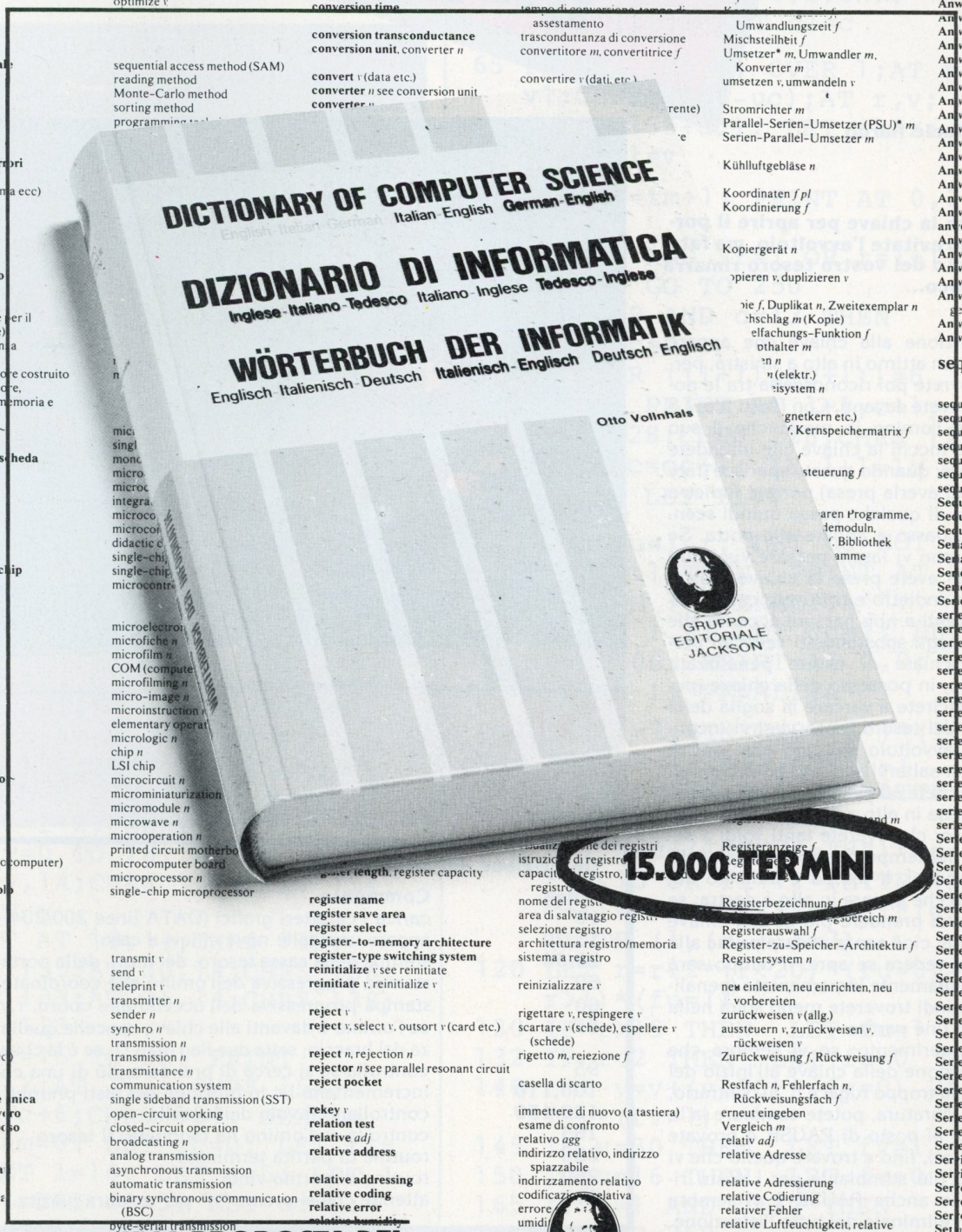
Libri e riviste di elettronica e informatica.



CORSI CONTINUI
TUTTO L'ANNO
CON I MIGLIORI
SOFTERISTI

IL TELEFONO È 02-437.385

L'ESIGENZA DI CAPIRE...



15.000 TERMINI

UNA PROPOSTA DEL GRUPPO EDITORIALE JACKSON

Pag. 920 Formato 18 x 25 cm. Cod. 100 H L. 55.000 (Abb. L. 44.000)

oscillante, ordinamento ~ — oscillating sort
oscillatore *m* — oscillator *n*
oscillatore a battimenti — beat frequency oscillator
oscillatore a quarzo — quartz oscillator
oscillatore a rilassamento — astable multivibrator
oscillatore di rilassamento a bloccaggio — blocking oscillator
oscillatore quarzato — quartz oscillator
oscillazione *f* — oscillation *n*
oscillazione *f* **a campo ritardante** — retarding-field oscillation
oscillazione pendolare — hunting *n* (unstable condition)
oscillografo *m* — oscillograph *n*
oscillografo a raggi catodici — cathode-ray oscilloscope (CRO)
oscillogramma *m*
oscilloscopio *m*
ospite, calcolatore ~
ottale *agg* — 8-bit byte
ottale, cifra ~
ottetto *m* (byte composto da 8 bits)
ottica delle fibre — fiber optics
ottico *agg* — optically readable
ottimare *v* — optimize *v*

metodo

metodo di accesso sequenziale — sequential access method (SAM)
metodo di lettura — reading method
metodo di Monte Carlo — Monte-Carlo method
metodo di ordinamento — sorting method
metodo di programmazione — programming
metodo di registrazione
metodo di ricerca
metodo di riconoscimento errori
metodo di trasmissione
mettere a punto (un programma ecc)
mettere a punto in linea
mezza parola
mezzi *m pl*
mezzo *m*
mezzo addizionatore
mezzo di immagazzinamento
mezzo di protezione dati
mezzo fisico di trasmissione
mezzo trasmissivo (hardware per il controllo della trasmissione)
MF = modulazione di frequenza
micro *m*
microcalcolatore *m* (calcolatore costruito intorno ad un microprocessore, generalmente completo di memoria e periferiche)
microcalcolatore, sistema a ~
microcalcolatore monochip
microcalcolatore su singola scheda
microciclo *m*
microcircuito *m*
microcircuito integrato
microcomputer *m*
microcomputer, piastra di ~
microcomputer didattico
microcomputer single chip
microcomputer su un unico chip
microcontroller *m*
microelaboratore *m* (raro) v. microprocessore
microelettronica *f*
microfiche *f*
microfilm *m*
microfilm, uscita su ~
microfilmatura *f*
microfotogramma *m*
microistruzione *f*
microistruzione *f*
micrologica *f*
micrologico *m*
micrologico LSI
microminiaturizzato, circuito ~
microminiaturizzazione *f*
micromodulo *m*
microonde *f*
microoperazione *f*
microplacchetta *f*
microplacchetta *f* (di un microcomputer)
microprocessore (μP) *m*
microprocessore a chip singolo

trasmettere

trasmettere *v* — transmit *v*
trasmettere *v* — send *v*
trasmettere per telescrivente — teleprint *v*
trasmettitore *m* — transmitter *n*
trasmettitore *m* — sender *n*
trasmettitore sincrono — synchro *n*
trasmissione *f* (di dati ecc.) — transmission *n*
trasmissione *f* (di messaggi ecc) — transmission *n*
trasmissione, fattore di ~ — transmittance *n*
trasmissione, sistema di ~ — communication system
trasmissione a banda laterale unica — single sideband transmission (SST)
trasmissione a corrente di lavoro — open-circuit working
trasmissione a corrente di riposo — closed-circuit operation
trasmissione a più terminali — broadcasting *n*
trasmissione analogica — analog transmission
trasmissione asincrona dei dati — asynchronous transmission
trasmissione automatica — automatic transmission
trasmissione binario-sincrona — binary synchronous communication (BSC)
trasmissione byte-seriale — byte-serial transmission
trasmissione circolare — broadcasting
trasmissione dati — data transmission
trasmissione dati asincrona — asynchronous transmission
trasmissione dati in start-stop — start-stop data transmission
trasmissione dati sincrona — synchronous data transmission
trasmissione dati su canale unico — single-channel data transmission
trasmissione dati via cavo telefonico — telephone line transmission
trasmissione dei dati — data traffic
trasmissione dei dati a distanza — remote data transmission
trasmissione di dati — data transmission
trasmissione di immagini — picture transmission
trasmissione di messaggi — message transmission
trasmissione diretta — direct transmission
trasmissione diretta dei dati — on-line data transmission
trasmissione in burst mode — burst mode (transmission)
trasmissione in corrente continua
trasmissione in fonia — voice communication
trasmissione in serie — serial transmission

initialize *v* (tape etc.) — inizializzare *v* (nastro ecc.) — vorbereiten *v*, etikettieren *v*
initialize *v* (program) — inizializzare *v* (programma) — vorbereiten *v*
initialize, to ~ a volume — inizializzare un volume — einen Datenträger einrichten, einen

conversion

conversion *n* (from old system to new one etc.) — trasformazione *f* (di un impianto ecc.) — Umrüstung *f*, Umstellung *f*
conversion (of signals etc.)
conversion, program ~ — trasformazione *f*, conversione *f* — Umformung *f*, Umsetzung *f*
conversion instruction — conversione di programmi — Programmkonvertierung *f*
conversion — istruzione di conversione — Umsetzbefehl *m*
conversion routine see translation program
conversion table, translation table — tabella di conversione — Umsetztabelle *f*, Umsetzungstabelle *f*
conversion time — assestamento — Umwandlungszeit *f*
conversion transconductance — trasconduttanza di conversione — Mischsteilheit *f*
conversion unit, converter *n* — convertitore *m*, convertitrice *f* — Umsetzer* *m*, Umwandler *m*, Konverter *m*
convert *v* (data etc.) — convertire *v* (dati, etc.) — umsetzen *v*, umwandeln *v*
converter *n* see conversion unit
converter *n* — Stromrichter *m*
Parallel-Serien-Umsetzer (PSU)* *m*
Serien-Parallel-Umsetzer *m*
Kühlluftgebläse *n*
Koordinaten *f pl*
Koordinierung *f*
Kopiergerät *n*
kopieren *v*, duplizieren *v*
Kopie *f*, Duplikat *n*, Zweitexemplar *n*
Kopierschlag *m* (Kopie)
Kopierfachungs-Funktion *f*
Kopierhalter *m*
Kopierer *n*
Kopierer (elektr.)
Kopiersystem *n*
Kopierstift etc.)
Kernspeicher, Kernspeichermatrix *f*
Kopiersteuerung *f*
Kopieren Programme, Kopiermodule, Bibliothek
Kopierprogramme
Kopierbereich *m*
Registeranzeige *f*
Registerbereich *m*
Registerbezeichnung *f*
Registerbereich *m*
Registerauswahl *f*
Register-zu-Speicher-Architektur *f*
Registersystem *n*
neu einleiten, neu einrichten, neu vorbereiten
zurückweisen *v* (allg.)
aussteuern *v*, zurückweisen *v*, rückweisen *v*
Zurückweisung *f*, Rückweisung *f*
Restfach *n*, Fehlerfach *n*, Rückweisungsfach *f*
erneut eingeben
Vergleich *m*
relativ *adj*
relative Adresse
relative Adressierung
relative Codierung
relativer Fehler
relative Luftfeuchtigkeit, relative Feuchtigkeit
Programmierung
Relais* *n*
Relaisrechner *m*
Relaiskoppelfeld *n*
freigeben *v* (Gerät, Puffer etc.)
Freigabe *f*
Zuverlässigkeit *f*, Betriebssicherheit
geschätzte Zuverlässigkeit

register length, register capacity — capacità di registro, registro
register name — nome del registro
register save area — area di salvataggio registri
register select — selezione registro
register-to-memory architecture — architettura registro/memoria
register-type switching system — sistema a registro
reinitialize *v* see reinitiate
reinitiate *v*, reinitialize *v* — reinizializzare *v*
reject *v* — rigettare *v*, respingere *v*
reject *v*, select *v*, outsort *v* (card etc.) — scartare *v* (schede), espellere *v* (schede)
reject *n*, rejection *n* — rigetto *m*, reiezione *f*
rejector *n* see parallel resonant circuit
reject pocket — casella di scarto
rekey *v* — immettere di nuovo (a tastiera)
relation test — esame di confronto
relative *adj* — relativo *agg*
relative address — indirizzo relativo, indirizzo spiazzabile
relative addressing — indirizzamento relativo
relative coding — codificazione relativa
relative error — errore relativo
relative humidity — umidità relativa
relative programming — programmazione relativa, programmazione in relativo
relay *v* see retransmit
relay *n* — relè *m*, relais *m*
relay calculator — calcolatore a relais
relay matrix — campo di accoppiamento relè
release *v* — liberare *v*, abilitare *v* (unità ecc.), rilasciare *v* (buffer ecc.)
release *n* — liberazione *f*, rilascio *m* (di unità ecc.)
release signal see enable signal
reliability *n* — affidabilità *f*
reliability, assessed ~ — affidabilità valutata
reliability, hardware ~ — affidabilità dell'hardware

Antivalenzglied *n*
Antrieb *m*
Antriebskette *f*
Antriebsmotor *m*
Antriebsregelung *f* (der Magnetbandeinheit)
Antriebswelle *f*
Antwort *f*
Antwort des Operators
antworten *v*
Antwortmeldung *f*
Antwortsignal *n*
Antwortzeit *f*
Anweisung* *f*
Anweisung *f*
Anweisung *f* (zur Gerätebe
Anweisung *f* (bei problemo Programmiersprachen)
Anweisung *f* (bei maschine Sprachen)
Anweisung an das Progran
Anweisung in Primärsprac
Anweisung, arithmetische
Anweisung, ausführbare ~
Anweisung, nichtausführb
Anweisung, symbolische ~
Anweisung, unbedingte ~
Anweisung, zusammenges
Anweisungsmarke *f*
Anweisungsnummer *f*
Anwender *m*
Anwenderarbeitsbereich *n*
Anwenderdatei *f*
Anwenderebene *f*
Anwendermakro *m*
Anwendermaske *f*
anwenderorientiert *adj*
Anwenderprogramm *n*
Anwenderprogrammierun
Anwenderschnittstelle *f*
Anwendersoftware *f*
Anwendersoftware *f* (vom geschrieben)
Anwendersysteme, dedizie
Anwenderunterprogramm

sequentielle

sequentielle Steuerung
sequentielle Struktur
sequentielle Verarbeitung
sequentielle Zugriffsmeth
sequentieller Rechner
sequentieller Speicher
sequentieller Zugriff
Sequentiellrechner *m*
Sequenz *f* (Sätze oder Zeic
Sequenzspeicher *m*
Serialdrucker *m*
Serialdruckwerk* *n*
Serie *f*
Serie *f*
Serie, in ~
seriell *adj* (Übertragung etc
seriell *adv*
seriell aufgebaut
serielle asynchrone Schnit
serielle Binärübertragung
serielle Organisation
serielle Start-Stopp-Übert
serielle Synchronübertragu
serielle Übertragung
serielle Verarbeitung
serieller Betrieb*
serieller Datentransfer
serieller Ein-/Ausgabekan
serieller I/O-Port
serieller Port
serieller Zugriff
Serienabtastung *f*
Serienaddition *f*
Serienausgang *m*
Serienbetrieb *m*
Serendrucker *m*
Serieneingang *m*
Serienmultiplikation *f*
Serennummer *f*
Serien-Parallelbetrieb *m*
Serien-Parallel-Schaltung
Serien-Parallelsystem *n*
Serien-Parallel-Umsetzer
Serien-Parallel-Umsetzun
Serienprogrammierung *f*
Serienrechner *m*
Serienschaltung *f*
Serienschnittstelle *f*
Serien-Serien-Betrieb *m*
Serienspeicher *m*
Serienspeicherung *f*
Serientransfer *m*
Serienübertragung *f*
Serienverarbeitung *f*
Service-Rechenzentrum *n*
Service-Techniker *m*
Servomechanismus *m*
Servomotor *m*
Servosystem *n*
Set *n*

grafische

grafische Einheit (normierte
grafische Lösung
grafisches Ausgabegerät
grafisches Datenverarbeitu
grafisches Symbol
grafisches Terminal
Grammatik *f*
grammatische Analyse
grammatischer Fehler
Graph *m*
Graphik... s. Grafik...
graphisch *adj* s. grafisch
Gray-Code *m*

La chiave magica

Load "chiave"

di **Raffaele Ilardo**

Trovate la chiave per aprire il portone ed evitate l'avvoltoio, ma fate presto o del vostro tesoro rimarrà ben poco...

Attenzione alla chiave che appare per un attimo in alto a sinistra, perché dovrete poi riconoscerla tra le nove che avete davanti. Con i tasti 5,6,7,8 guidate l'omino in modo che il suo braccio tocchi la chiave che intendete prendere; quando questa sparisce (fate conto di averla presa) portate indietro l'omino di qualche passo, quindi scendete in basso e andate alla porta. Se questa non vi lascia passare, vuol dire che non avete preso la chiave giusta. Tornate indietro e riprovate con un'altra. Attenti a non passare accanto alle chiavi negli spostamenti verticali perché rischiate di essere penalizzati. Quando, in possesso della chiave giusta, riuscirete a varcare la soglia della stanza del tesoro, ecco venirvi incontro un avvoltoio stregato, che farà di tutto per saltarvi addosso ed uccidervi. Se riuscirete ad evitarlo e a raggiungere la cassa in alto a destra, il tesoro sarà vostro: ci troverete tanti soldi a seconda del tempo che avete impiegato per arrivarci. Il tempo è indicato dal numero che appare in alto a destra: se cercate di prendere più di una chiave per volta, cioè senza andare prima alla porta a vedere se apre, il tempo sarà istantaneamente incrementato (penalità) e quindi troverete meno soldi nella cassa a fine partita.
Un suggerimento: se vi sembra che l'apparizione della chiave all'inizio del gioco sia troppo fugace o, al contrario, troppo duratura, potete cambiare la linea 20: al posto di PAUSE 4 provate altri valori, fino a trovare quello che vi sembra più stimolante per i vostri riflessi; se anche PAUSE 1 vi sembra troppo, eliminate del tutto l'istruzione.

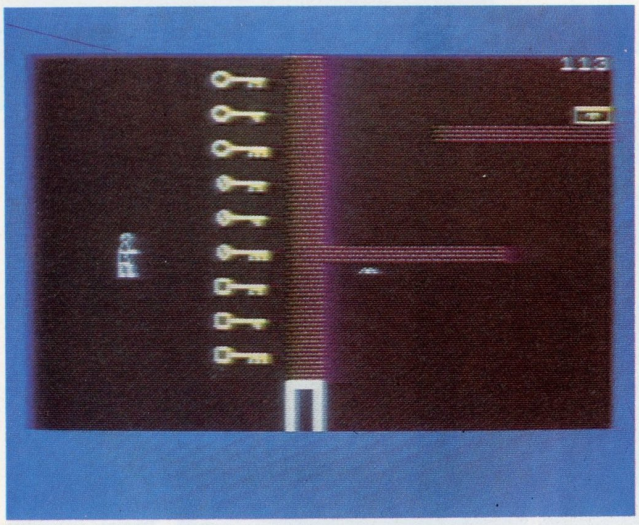
Descrizione del programma

Linee	Commento
10	carica i caratteri grafici (DATA linee 200/204)
15,20	appare una delle nove chiavi a caso
25/50	stampa della cassa tesoro, dei muri, della porta in basso
60	stampa progressiva dell'omino alle coordinate l,c
60	stampa progressiva dell'uccello alle coord. r,v
85	se l'omino è davanti alle chiavi, cancella quella all'altezza del braccio; setta due flag (flag f1, se è la chiave giusta)
95	controlla se si cerca di prendere più di una chiave
100,110	incrementano l,c in funzione dei tasti premuti
120/150	controllano il volo dell'uccello
165	controlla se l'omino ha raggiunto il tesoro
250	routine di partita terminata
260	routine di partita vinta
310	attende che si prema "n" per un'altra partita.

La chiave magica versione per ZX Spectrum 16K/48K

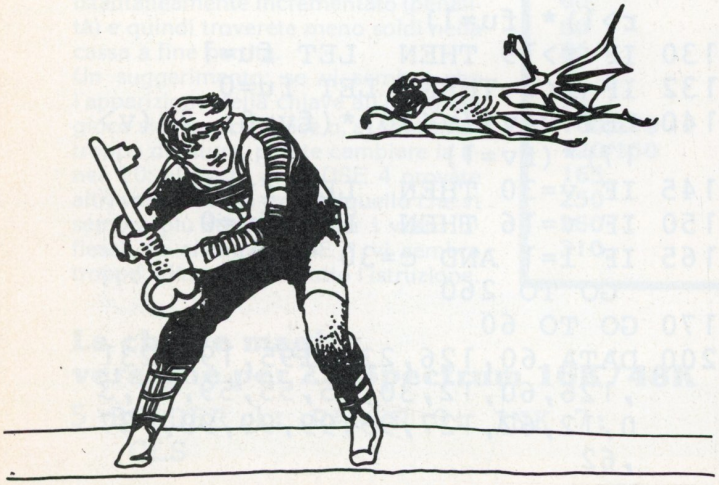
```
5 BORDER 1: PAPER 0: INK 7:
  CLS
10 FOR n=USR "a" TO USR "o"+7:
   READ d: POKE n,d: NEXT n
15 LET t=144+INT (RND*3):
   LET d=147+INT (RND*3)
```

La chiave magica

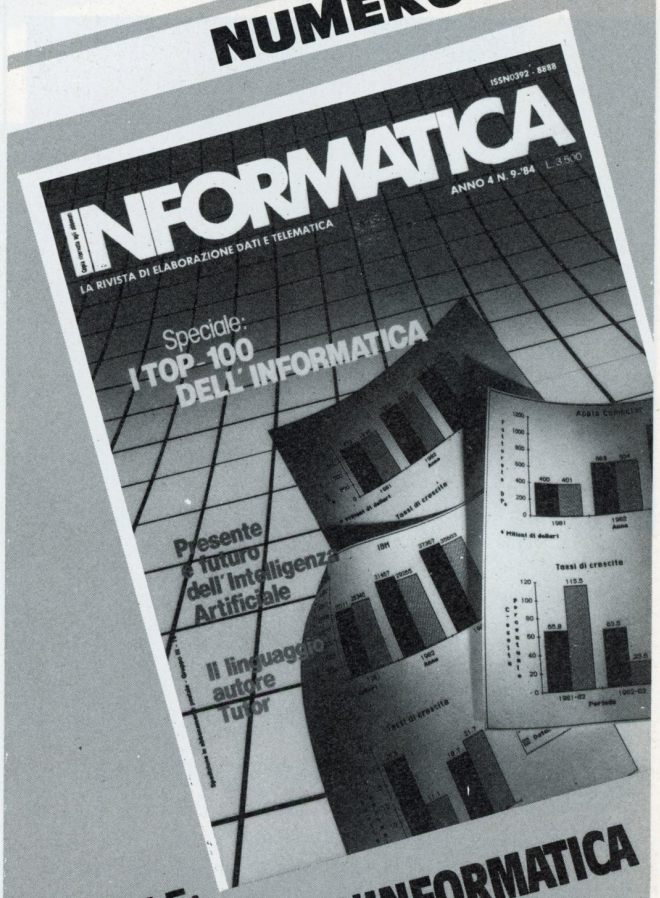



```
20 PRINT  INK 6;AT 5,5;CHR$ t;
   CHR$ 150;CHR$ d: PAUSE 4:
    CLS
25 PRINT  INK 6;AT 3,30;CHR$ 1
   57;CHR$ 158
30 LET l=1: FOR a=144 TO 146:
    FOR b=147 TO 149: PRINT
    INK 6;AT 1,10;CHR$ a;
   CHR$ 150;CHR$ b: IF a=t
    AND b=d THEN  LET lc=l
35 LET l=l+2: NEXT b: NEXT a
40 FOR y=0 TO 18: PRINT  INK 2
   ;AT y,14;CHR$ 154;CHR$ 154:
    NEXT y
45 PRINT AT 19,14;CHR$ 139;
   CHR$ 135;AT 20,14;CHR$ 138;
   CHR$ 133;AT 21,14;CHR$ 138;
   CHR$ 133
50 FOR c=16 TO 25: PRINT
    INK 2;AT 11,c;CHR$ 154;
   AT 4,c+6;CHR$ 154: NEXT c
55 LET tm=0: LET cc=0: LET fl=
   0: LET l=10: LET c=5: LET l
   l=10: LET cl=5: LET uc=0:
    LET fv=0: LET fu=0: LET r=
   0: LET rl=0: LET v=30:
    LET vl=30
57 PRINT AT 0,30; INK 8;
    OVER 1;CHR$ 156: PRINT
    OVER 1;AT 10,5;CHR$ 151;
   AT 11,5;CHR$ 152;AT 12,5;
   CHR$ 153
60 PRINT  INK 8; OVER 1;AT 11,
   cl;CHR$ 151;AT 11+1,cl;
   CHR$ 152;AT 11+2,cl;CHR$ 15
   3;AT l,c;CHR$ 151;AT l+1,c;
   CHR$ 152;AT l+2,c;CHR$ 153:
    LET 11=l: LET cl=c
65 PRINT  INK 8; OVER 1;AT rl,
   vl;CHR$ (156-uc);AT r,v;
   CHR$ (155+uc): LET rl=r:
    LET vl=v
70 LET tm=tm+1: PRINT AT 0,29;
   tm
75 IF v=c AND (r=l OR r=l+1)
    THEN  GO TO 250
80 IF l=19 AND c=13 THEN
    LET cc=0
85 IF ATTR ((l+1),(c+1))=6
    THEN  PRINT AT l+1,c+1;
   CHR$ 128;CHR$ 128;CHR$ 128:
    LET cc=cc+1: IF l+1=lc
    THEN  LET fl=1
90 LET fu=l<r: LET uc=NOT uc
95 IF cc>1 THEN  PRINT AT 0,0;
    FLASH 1;"PENALITA'":
    BEEP 1,8: LET cc=1: LET tm
   =tm+100: PRINT AT 0,0;
    OVER 1;"PENALITA'"
100 LET l=l+(INKEY$="6")*(l<19)
    *(ATTR ((l+3),c)=7)-(
    INKEY$="7")*(l>0)*(ATTR ((l
    -1),c)=7)
110 LET c=c+(INKEY$="8")*(c<31)
    *(c<13 OR fl=1)*(ATTR (l+1,
    c+1)=7)-(INKEY$="5")*(c>0)*
    (ATTR (l,(c-1))=7)
120 LET r=r+2*(r<20)*(fu=0)-2*(
    r>1)*(fu=1)
130 IF r>19 THEN  LET fu=1
132 IF r<2 THEN  LET fu=0
140 LET v=v+(v<30)*(fv=0)-2*(v>
    17)*(fv=1)
145 IF v=30 THEN  LET fv=1
150 IF v=16 THEN  LET fv=0
165 IF l=1 AND c=30 THEN
     GO TO 260
170 GO TO 60
200 DATA 60,126,231,195,195,231
    ,126,60,12,30,55,55,59,59,3
    0,12,62,127,99,99,99,99,127
    ,62
```

```
201 DATA 0,0,0,254,254,252,204,
    204,0,0,0,254,254,120,56,24
    ,0,0,0,255,255,255,219,219
202 DATA 0,0,0,255,255,0,0,0,0,
    0,0,56,104,100,72,48,80,72,
    136,207,255,144,144,136,136
    ,248,220,204,204,204,238,23
    8
203 DATA 0,221,0,119,0,221,0,11
    9,0,0,0,36,90,153,0,0,0,129
    ,66,36,24,24,0,0
204 DATA 255,255,128,135,133,12
    9,128,255,255,255,1,225,161
    ,129,1,255
250 FOR m=0 TO 8: FOR n=7 TO 1
     STEP -1: BORDER n: PAUSE 1
    : NEXT n: NEXT m: BEEP .1,-
    5: BEEP .1,-5: BEEP .1,-5:
     BEEP 1,-9: PAUSE 100:
     GO TO 300
260 RESTORE 280: PRINT AT 20,4;
    "HAI RAGGIUNTO IL TESORO";
    AT 21,1;"IL TUO PREMIO E' D
    I";CHR$ 128;INT (100000/tm)
    ;CHR$ 128;"SCUDI"
270 FOR t=1 TO 3: PAUSE 15:
     FOR s=1 TO 4: READ d,a:
     BEEP d,a: NEXT s:
     RESTORE 280: NEXT t
280  DATA .2,0,.2,4,.2,5,.4,7
290 BEEP .4,4: BEEP .4,0:
     BEEP .4,4: BEEP .4,2
300 PRINT AT 11,2;"premi N per
    un'altra partita"
310 IF INKEY$<>"n" THEN
     GO TO 310
320 CLS : PAUSE 100: GO TO 15
```



E' IN EDICOLA
IL NUOVO
NUMERO DI



SPECIALE:
I TOP 100 DELL'INFORMATICA

Presente e futuro dell'Intelligenza artificiale

Il Linguaggio Tutor

Una pubblicazione firmata...

GRUPPO EDITORIALE JACKSON s.r.l
20124 Milano - Via Rosellini, 12 - Tel. 02/68.80.951-2-3-4-5
Telex: 333436 GEJ IT

ZX CLUB

I Club Sinclair

Molta carne al fuoco per questa puntata di ZX Club, soprattutto per ciò che riguarda annunci e comunicazioni di ogni tipo. Ci facciamo quindi subito da parte per cedere la parola a singoli ed associazioni!

Gli annunci di ZX Club

BIT & BYTES CAPO D'ORLANDO COMPUTER CLUB annuncia la propria nascita, avvenuta in data 1/8/1984. Così recita l'art. 1 dello statuto del club: "Il club assume quale fine principale della propria attività la diffusione massima degli strumenti di padronanza dei rapporti tra individui e computer."
Gli interessati sono invitati a contattare il club per dare vita alle iniziative ritenute più opportune. Ogni contributo di idee, impegno e aiuto economico è gradito.
L'indirizzo del club è
BIT & BYTES CAPO D'ORLANDO COMPUTER CLUB
c/o Giuseppe RICCIARDI
Via Vittorio Veneto, 56
98071 Capo d'Orlando ME
Tel. 0941/901670

Compro vendo, cambio software per ZX Spectrum e Commodore 64. Scrivete ed entrerete a fare parte del club NEW INFORMATICA.
CLUB NEW INFORMATICA
c/o Rocco Luca MAROTTA
Via Federico Persico, 3/E
80141 Napoli NA
Tel. 081/7807818

È sorto un nuovo club: il club U.S.A. (Utilizzatori Spectrum all'Avanguardia). Il suo scopo è di agevolare lo scambio di software, senza alcun fine di lucro. Per informazioni scrivere (allegando un francobollo per la risposta) a
CLUB U.S.A.
Via Sandro Botticelli, 17
50018 Scandicci FI
Tel. 055/254808

Il Sir Clive Club scambia idee e software originale per Spectrum e QL e ricerca futuri possessori di QL per primo contatto. Se possibile, allegate il bollo per la risposta, in ogni caso rapida e sicura.
SIR CLIVE CLUB
c/o Roberto GHEZZI
Via Volontari del Sangue 202
20099 Sesto S. Giovanni MI

Attenzione! Sono aperte le iscrizioni al GSCC. La quota, fissata a lire 15000 per il 1985, dà diritto a ricevere bimensilmente il bollettino (di 20/30 pagine circa) e a partecipare alle moltissime attività del Sinclair Club. Per iscrizioni e chiarimenti:
GSCC
c/o Simone PICCINI
Via C. Pisacane, 9
50015 Grassina FI

Siamo diversi ragazzi di Bari che hanno costituito un club per utenti Sinclair. Abbiamo mille idee e centinaia di programmi. Chiunque può aderire, vi aspettiamo! Scrivete presso
Giuseppe SETTANNI
Via Murat, 86
70100 Bari BA
Tel. 080/215365

Cerco sinclairisti per scambio idee e programmi, solo a Pesaro.
Marino RUGGERI
Via Milazzo, 17
61100 Pesaro PS
Tel. 0721/452611

Contatterei possessori di Spectrum per scambio di software ed esperienze.
Roberto PELLACANI
Via Matteotti, 51
46025 Poggio Rusco MN

Compro o scambio, a basso prezzo, le cassette "Alchemist", "Chequered Flag", "Ant Attack", solo in Milano. Cerco inoltre Sinclair Club, sempre a Milano.
Telefonate!
Bruno COVER
V.le D. Ranzoni, 1
20149 Milano MI
Tel. 02/4033117

Sto fondando un club per possessori di TI99/4A e ZX Spectrum e cerco collaboratori disposti a mandare software e hardware per questi computer.
Ogni collaboratore riceverà ogni mese tutti i programmi del club, con istruzioni dettagliate. Il club vende anche programmi bellissimi a prezzi bassissimi: richiedete il listino.
Indicate un gioco da bar che vi piace particolarmente: dopo alcuni mesi lo vedrete sul vostro Spectrum o, più difficilmente, sul TI.
Cerco infine collaboratori che sappiano convertire i programmi in diversi linguaggi.
Ettore CARPINELLA
V.le Europa 4
83034 Casalbore AV
Tel. 0825/849281

In data 20/2/1984 si è costituito a Roma il SINCLAIR CLUB ROMA-TALENTI. Il club è una associazione senza scopi di lucro ed a tal fine l'iscrizione è completamente gratuita. Gli scopi principali del club sono:
1) mettere in contatto tra loro tutti gli appassionati di computer Sinclair;
2) offrire a tutti i soci la possibilità di abbonarsi a "Sperimentare con l'elettronica e il computer" a prezzo ridotto (circa il 30% in meno);
3) acquisire una valida raccolta di software per lo Spectrum;
4) mettere questa raccolta a disposizione di tutti i soci. Per usufruire del software del club i soci si devono impegnare ad ampliare la raccolta con ulteriori programmi di buon livello;
5) acquistare periodicamente, tramite una raccolta di fondi tra tutti i soci in-

Per collaborare a SUPERSINC

La maggior parte dei seguenti suggerimenti ha lo scopo di migliorare l'accuratezza e la velocità di pubblicazione di un articolo; rispettando questi consigli si accresceranno le probabilità che un vostro lavoro venga pubblicato. La rivista è interessata ad articoli e programmi riguardanti la linea Sinclair. Siamo più interessati al contenuto di un articolo piuttosto che al suo stile, e soprattutto gli articoli devono essere chiari ed esaurienti. La seguente guida permetterà che le vostre buone idee e i vostri programmi vengano più facilmente accettati per la pubblicazione:

1 l'angolo superiore sinistro della prima pagina dovrà contenere: nome, cognome, indirizzo, numero telefonico, codice fiscale e data di spedizione, luogo e data di nascita.

2 l'angolo superiore destro della prima pagina dovrà contenere la marca e il tipo di computer al quale il lavoro si riferisce, unitamente alla configurazione richiesta (memoria occorrente, eventuali periferiche e così via).

3 il titolo sottolineato dell'articolo dovrà iniziare a circa due terzi in altezza della prima pagina.

4 le pagine seguenti potranno essere battute normalmente, con la condizione che l'angolo superiore destro contenga un'abbreviazione del titolo e del cognome, unitamente al numero di pagina progressivo. Per esempio, Horace Goes.../Brambilla/2.

5 tutte le linee del testo dell'articolo dovranno essere battute con spazio 2 o spazio 3, e un margine di circa un centimetro dovrà trovarsi ad entrambi i lati dello scritto.

6 dovrà essere usata una carta formato A4 e lo scritto dovrà occupare un solo lato del foglio (caratteri maiuscoli e minuscoli).

7 i fogli dovranno essere uniti con una clip.

8 avendo intenzione di spedire più di un articolo, questi dovranno essere inviati separatamente insieme alla rispettiva copia su supporto magnetico.

9 programmi brevi (meno di 20 linee) protranno essere inseriti nel testo, mentre programmi più lunghi dovranno essere listati separatamente. È **ESSENZIALE** per noi disporre di una copia del programma registrata più volte su supporto magnetico, su entrambi i lati dello stesso. È preferibile usare nastri di buona qualità e di lunghezza non eccessiva; la cassetta o la cartuccia per Microdrive dovranno essere etichettati con il nome dell'autore, il titolo dell'articolo, il computer interessato e soprattutto le eventuali espansioni richieste. Come suggerimenti di programmazione, si consiglia di usare, per esigenze di stampa listati, le istruzioni INK, PAPER, INVERSE piuttosto che scrivere direttamente in INVERSE VIDEO. Un rapido controllo dei programmi per operare queste sostituzioni sarà da noi estremamente apprezzato

10 per maggior chiarezza, all'interno dell'articolo è conveniente usare caratteri maiuscoli riferendosi a istruzioni BASIC (esempio RETURN, LIST, RND, PRINT etc.). Se si desidera evidenziare una parola, è preferibile sottolinearla piuttosto che scriverla in carattere maiuscolo.

11 gli articoli ed i programmi potranno avere qualsiasi lunghezza — da una routine di una sola linea fino a programmi molto complessi.

12 volendo includere fotografie, questi dovranno essere formato 24 × 36, o 6 × 6, in bianco e nero o diapositive.

13 non prenderemo in considerazione articoli che siano stati sottoposti ad altre case editrici.

14 il compenso per la collaborazione prestata sarà commisurato alla complessità e all'interesse del programma (da un minimo di L. **50.000** a un massimo di L. **300.000**). Il pagamento è effettuato in caso di pubblicazione del lavoro.

15 il materiale ricevuto e non pubblicato non verrà restituito.

Spedite i vostri lavori a:
SUPERSINC
Via Rosellini, 12
20124 Milano
e saremo lietissimi di pubblicare i contributi migliori.

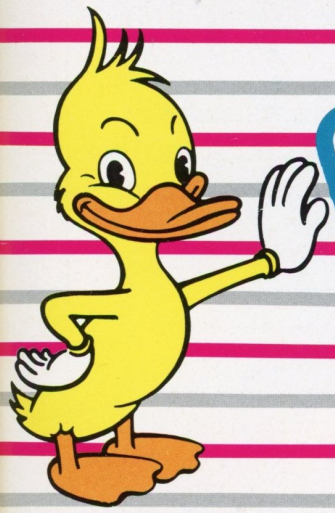
La Redazione

IL PRIMO SETTIMANALE DI SOFTWARE SU CARTA

L. 1.000

PER IL TUO PERSONAL COMPUTER

Una pubblicazione della J.*soft* editrice



PAPER soft

Spedi. in abb. post. gr. II/70

apple

TI-99/4A

sinclair Spectrum

C64

VIC 20

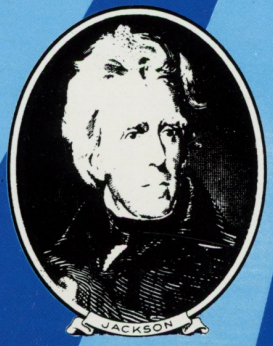
In edicola ogni venerdì!

Editrice J.soft via Rosellini, 12 - 20124 Milano - tel. 02/6888228 - 683797 - 6880841 - 6880842 - 6880843

PERSONAL COMPUTER: le prime, le migliori!

PERSONAL SOFTWARE:
L'unica che presenta software per tutti i personal: Commodore, Apple, Sinclair, T.I., HP, Sharp, Sega, Olivetti, ecc.

BIT La più letta,
la prima e più diffusa.
TEST: SPECTRAVIDEO SVI-728
SUPERBIT - 64 pagine di programmi
per i vostri personal computer.



Con tutta la competenza del

GRUPPO EDITORIALE

JACKSON